AFRL-IF-RS-TR-2005-36
Final Technical Report
February 2005

AUTONOMOUS NEGOTIATING TEAMS AND MODEL-INTEGRATED COMPUTING FOR AUTONOMIC LOGISTICS

Vanderbilt University

Sponsored by
Defense Advanced Research Projects Agency
DARPA Order No. J132

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE
ROME RESEARCH SITE
ROME, NEW YORK

STINFO FINAL REPORT

This report has been reviewed by the Air Force Research Laboratory, Information Directorate, Public Affairs Office (IFOIPA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

AFRL-IF-RS-TR-2005-36 has been reviewed and is approved for publication

APPROVED: /s/
RICHARD A. HYLE
Project Engineer

FOR THE DIRECTOR: /s/
JAMES A. COLLINS, Acting Chief
Advanced Computing Division
Information Directorate

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 074-0188

1. AGENCY USE ONLY (Leave blank)

2. REPORT DATE
February 2005

3. REPORT TYPE AND DATES COVERED
FINAL **May 99 – Jun 04**

4. TITLE AND SUBTITLE

AUTONOMOUS NEGOTIATING TEAMS AND MODEL-INTEGRATED COMPUTING FOR AUTONOMIC LOGISTICS

5. FUNDING NUMBERS
G - F30602-99-2-0505
PE - 62301E
PR - H360
TA - 01
WU - 01

6. AUTHOR(S)

Gabor Karsai
Benoit Dawant

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Vanderbilt University
Div. Sponsored Research
Station B #357749
Nashville TN 37235

8. PERFORMING ORGANIZATION REPORT NUMBER

N/A

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Defense Advanced Research Projects Agency
2701 North Fairfax Drive
Arlington VA 22203-1714

AFRL/IFTB
525 Brooks Road
Rome NY 13441-4505

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

AFRL-IF-RS-TR-2005-36

11. SUPPLEMENTARY NOTES

AFRL Project Engineer: Richard A. Hyle/IFSB/(315) 330-4857 Richard.Hyle@rl.af.mil

12a. DISTRIBUTION / AVAILABILITY STATEMENT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

12b. DISTRIBUTION CODE

13. ABSTRACT ***(Maximum 200 Words)***

The Model-Integrated Computing and Autonomous Negotiating Teams for Autonomic Logistics (MICANTS) project described in this report addressed the problem of building advanced decision support tools for maintenance planning and scheduling using negotiation technology. The project has developed and applied negotiation tools for aircraft maintenance planning and scheduling, and has demonstrated the feasibility of the approach in real-life examples: in AV8-B squadrons and in space launch preparation scheduling. The report summarizes the technological contributions of the project, and the actual prototype systems constructed.

14. SUBJECT TERMS
Decision support, aircraft maintenance scheduling

15. NUMBER OF PAGES
126

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

Table of Contents

List of Figures

1. Introduction

As stated in the eBook of the DARPA-IXO Autonomous Negotiating Teams (ANTS) project (http://www.isi.edu/~szekely/antsebook/ebook/), the goal of ANTS is to autonomously negotiate the assignment and customization of resources, such as weapons, to tasks, such as moving targets. To achieve this, systems need to be built that can operate effectively in highly decentralized environments, making maximum use of local information, providing solutions that are both good enough, and soon enough. These systems need to have components that communicate effectively with local peers, and also with information and command concentrators at higher levels of situation abstraction. They need to explicitly represent goals, values, and assessments of likelihood and assurance, and reason about those quantities and qualities in order to accomplish their mission. ANTs systems need to be designed to scale up and work efficiently on very large problems by making maximum use of localized, rather than global information, and by explicitly making decision theoretic trade-offs with explicit time-bounds on calculation of actions. This new technology will enable engineers to build systems that are designed to utilize, at the application level, all the distributed, networked computational resources (hardware, operating systems, and communication) that have been developed over the past two decades.

Maintenance Planning Agent's (MAPLANT) is the result of efforts conducted at Vanderbilt University under this program. The main objective of this project has been to explore the combined utilization of model-integrated computing and agent/negotiation technology to solve complex resource management problems in logistics. The long-term vision of the project is to integrate two technologies to develop and test a prototype information system for supporting aircraft logistic: model-integrated computing and autonomous negotiating teams. Model-integrated computing (MIC) technology is used to develop and evolve the basic capabilities and architecture of the information system. ANTS technology is used to address the issues of a distributed problem in the system. Model-integrated computing provides sophisticated modeling and system synthesis capabilities. ANT provides the underlying technology for light-weight components that "live" in the system and are involved in complex, distributed problem solving activities, which are exceedingly difficult to implement otherwise. The key ideas that have been developed and implemented in MAPLANT include dynamically evolving, negotiation-dependent preferences and values tradeoffs; rapid, incremental, and time-bounded negotiation processes; negotiation through incremental satisfaction of distributed constraints; and autonomous logistics realized through integration of legacy systems.

The potential impact of the system that has been developed is significant. It includes (1) improvement in efficiency of current maintenance logistic processes, e.g., increase in combat readiness and efficiency, decrease in cost, or reduced accident rate, (2) intelligible automated processes that support and facilitate distributed human decision, thus saving work and reducing human errors, (3) flexible approach that permits easy and rapid customization and adaptation to dynamic environments, and (4) the integration of existing and legacy systems into a cooperative environment.

Concepts and ideas have been reduced to practice and a close collaboration with end users produced a system that has been applied on the AV8-B under the Coherent Analytical Computing Environment (CACE) Advance Concept Technology Demonstration (ACTD). It has been field-tested at USMC MAG-13, in Yuma, AZ, and on several deployments. MAPLANT works together with other software tools developed under the CACE ACTD; notably Schedules Negotiated by Ant-based Planners (SNAP): a flight scheduler tool, and a system called Mission-Sensitive Aircraft Resume (MSAR) ---both developed by USC/ISI---, and a Data Warehouse (developed by LLD). The technology has also been transitioned to the United States Air Force's 45th Space Wing.

2. Technical Contributions

2.1 Overall system architecture

Figure 1 shows a high-level view of the overall MAPLANT architecture.

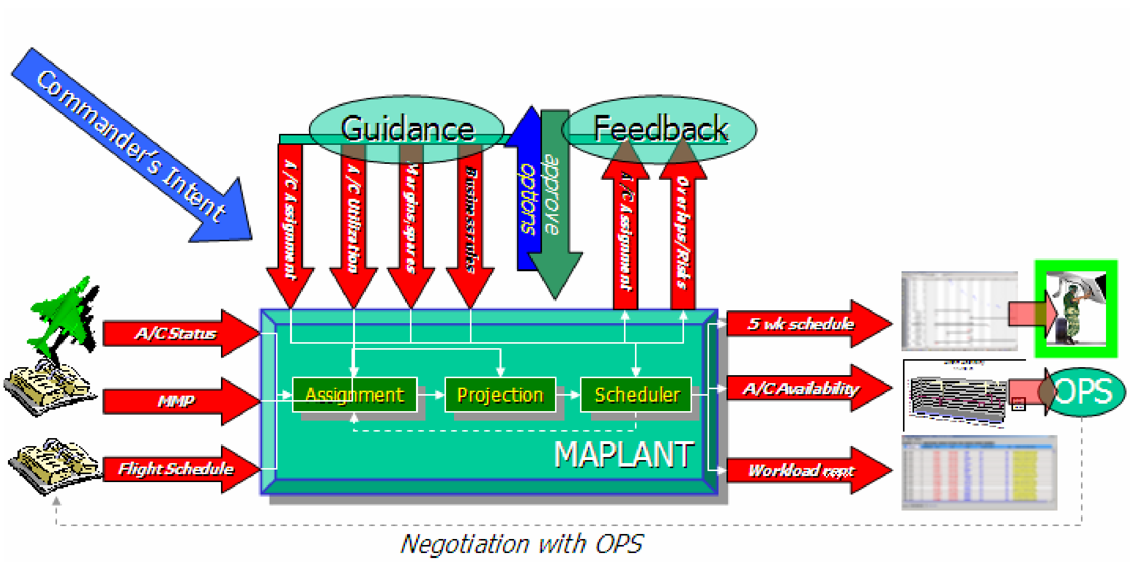


Figure 1: Overall architecture of MAPLANT

The system receives input from various data sources (e.g. legacy databases, on-line systems, etc.), allows a user to specify guidance, and generates (1) recommendations for aircraft to mission assignment, (2) prioritized lists of short-term (next-shift) maintenance actions to be performed, and (3) long-term (5+ weeks) maintenance schedules. The data input to MAPLANT includes: flight schedule for the planning period, current aircraft status of the squadron, open maintenance activities on aircraft, and upcoming calendar-based inspections. The user can set various guidance parameters on the system to influence the solutions to be generated. Guidance includes: particular aircraft-to-mission assignments, expected utilization of specific aircrafts over the month, spare aircraft policies, maintainer resource margins, and special business rules. Once the guidance is set, the user can query MAPLANT to generate recommendations for aircraft to mission assignments. While making the assignments, MAPLANT can interrogate Mission Sensitive Aircraft Resume (MSAR) that provides it with aircraft-specific information (e.g. weapon configuration or minor problems that do not impact on air worthiness but may prevent a plane to complete certain types of missions) to make the best choices for pairing up missions with aircrafts. Once the assignments are made, the user can review and fine-tune them. If the assignment results in new, high-priority maintenance tasks, MAPLANT uses them in the subsequent scheduling step. All assignments will result in putting flight hours on the aircraft, and this information is computed by MAPLANT to project upcoming, usage-based inspections. The user can analyze the risks associated with maintenance actions (with respect to manpower availability) and specify guidance on how to schedule those tasks. Once the guidance is specified, MAPLANT creates a detailed maintenance schedule that schedules all (1) upcoming short-term tasks, (2) calendar-based and (3) usage-based inspections, and which is guaranteed to satisfy all hard constraints (e.g. maintainer availability) and the maximum of soft constraints (e.g. preferences on scheduling certain activities simultaneously). The detailed maintenance schedule is visualized in Microsoft Project. Additional reports are generated for aircraft availability and upcoming workcenter activities.

2.2 Negotiation in MAPLANT

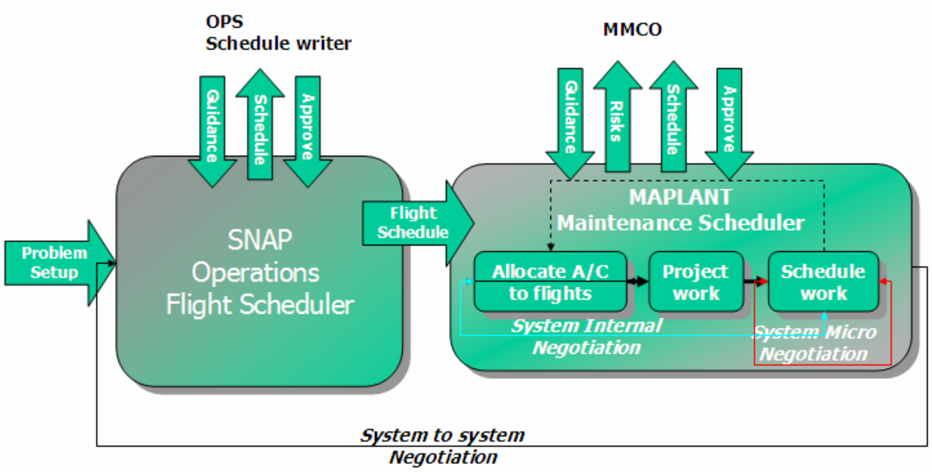


Figure 2: MAPLANT negotiates solutions at three levels

For complex tasks that involve a number of organizations and individuals as is the case for aircraft logistics, negotiation occurs at various levels. This is reflected in the various approaches that are used for negotiation in MAPLANT: system-to-system, internal but at the system level, and internal at the micro level as shown in Figure 2.

The main system with which MAPLANT interacts is SNAP developed by ISI. SNAP takes as input goals set up by OPS to satisfy a number of constraints ranging from supporting required missions to satisfying training needs. These goals are translated into desired flight schedules, i.e., number, type, and time of sorties for each day in the planning horizon. Flight schedules are passed to MAPLANT using a messaging protocol that has been designed to permit interaction between the systems. Flight schedules and maintenance/scheduling guidance provided by the MMCO (e.g., average monthly usage of aircraft, over- or under-utilization of specific aircraft to distribute major maintenance events uniformly over time) serve as input to MAPLANT. These are then translated into allocation of aircraft to flights, which is then translated into a maintenance schedule for each aircraft. As shown in Figure 2, negotiation happens between SNAP and MAPLANT. Flight schedules requested by SNAP may simply be impossible to satisfy with the available aircraft, personnel, or equipment. When this is the case, MAPLANT informs SNAP and provides it with information on missions and sorties that cannot be supported. SNAP modifies the flight schedule and resends it to MAPLANT. At this level, negotiation is achieved by means of a communication protocol that has been established between the two systems. This capability has been demonstrated. In the fall of 01 ISIS has successfully executed an integrated demonstration with three systems: MAPLANT, SNAP, and the data warehouse that were running in three, geographically different locations. In this experiment, MAPLANT and SNAP negotiated over the flight schedule. SNAP generated a flight schedule, informed MAPLANT about it, MAPLANT generated a maintenance schedule and aircraft availability reports and sent it back to SNAP for refining the flight schedule using it.

As shown in Figure 3, scheduling within MAPLANT occurs at two levels. The A-scheduler assigns aircraft to mission and the M-scheduler schedules the maintenance events once aircraft have been assigned to the missions. As discussed earlier, aircraft maintenance tasks fall into one of two broad categories: calendar-based and usage-based maintenance. The first category involves a series of routine maintenance tasks that need to be performed daily, weekly, monthly, etc. whether or not the aircraft flies. Usage-based maintenance tasks have to be performed after the plane has flown a predetermined number of hours (e.g., 56 hours). Certain inspections are long and require a substantial amount of manpower. To avoid finding himself understaffed, a good maintenance scheduler assigns aircraft to missions in such a way that long and labor-intensive maintenance are not performed on two planes at the same time.

MAPLANT follows the same approach. First, the A-scheduler assigns aircraft to missions. When this is done the number of flight hours for each plane can be computed (the mission specification includes its length). This, in turn, permits estimating requirements in terms of personnel and equipment resources needed to support the maintenance schedule. MAPLANT includes a resource monitor that detects time intervals over which resources are scarce and during which all maintenance events may not be supported. These are fed back to the A-scheduler that modifies airplane to mission assignments to permit support of all maintenance tasks. If a solution cannot be found MAPLANT informs SNAP.

The most complex component of MAPLANT is the M-scheduler. In its current state, MAPLANT is capable of scheduling over 3000 maintenance tasks performed by over 200 maintainers over a 5 week planning window with a time resolution of 15 minutes. It is scheduled in about 70 seconds on a standard P3/1.7GHz/512MB laptop. If the same schedule had to be done manually, this would require days of work. Early in the project ISIS relied on a SAT engine, which required expressing the scheduling problem as a set of Boolean CNF expressions. In late 2001, we changed our approach and decided to rely on a constraint programming system (called MOZART) based on the Oz language. The system supports concurrent constraint programming techniques, and allows the programming of high-performance search engines. It also has built-in libraries for finite domain, and finite set constraints, which can directly support the kind of problem solving MAPLANT requires. The negotiation happens at the domain-specific constraints level, and there is no need to transform the problem into a Boolean SAT problem first. We have experimented with a number of strategies,

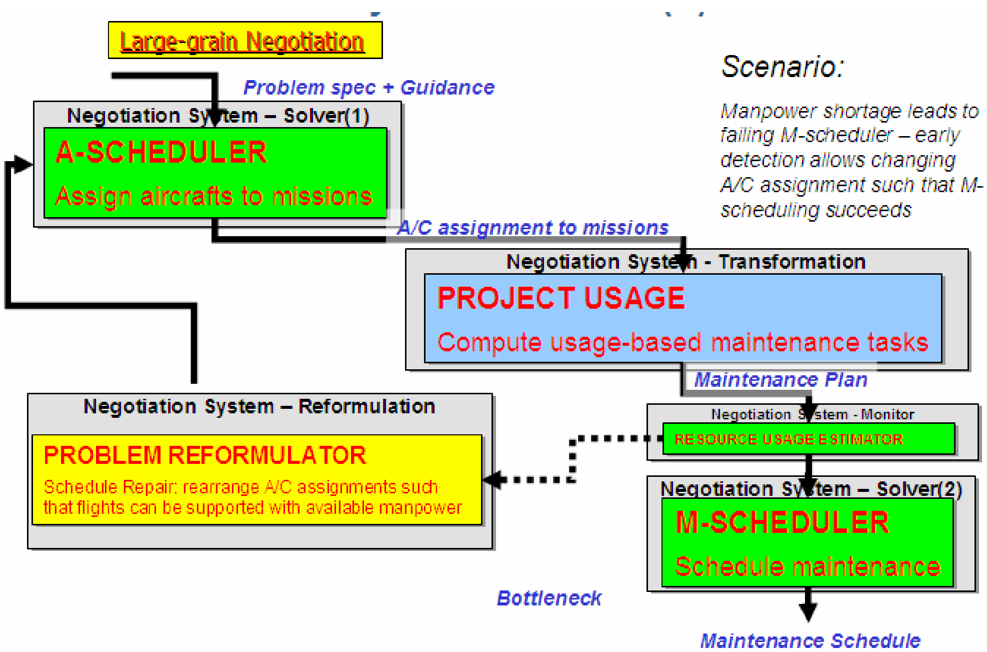


Figure 3: Interaction between the A-scheduler and the M-scheduler in MAPLANT

and ISIS has created a schedule generator engine that is compatible with the rest of the system. This led to a dramatic increase in performance and it allowed us to scale up the size of the problem. Another important aspect of the Oz-based implementation is that the search engine can be modified, which permits experimenting with various negotiation strategies as well as customization of these strategies. In particular, we have explored anytime strategies for negotiation. In this approach, the algorithm generates problem solutions while minimizing operational risks. It was shown that when this strategy was used, the system was able to generate increasingly better solutions as time went on

The two basic techniques of constraint programming are constraint propagation and constraint distribution. Constraint propagation is an efficient inference mechanism obtained with concurrent propagators accumulating information in a constraint store. Constraint distribution splits a problem into complementary cases once constraint propagation cannot advance further. By iterating propagation and distribution, propagation will eventually determine the solutions of a problem.

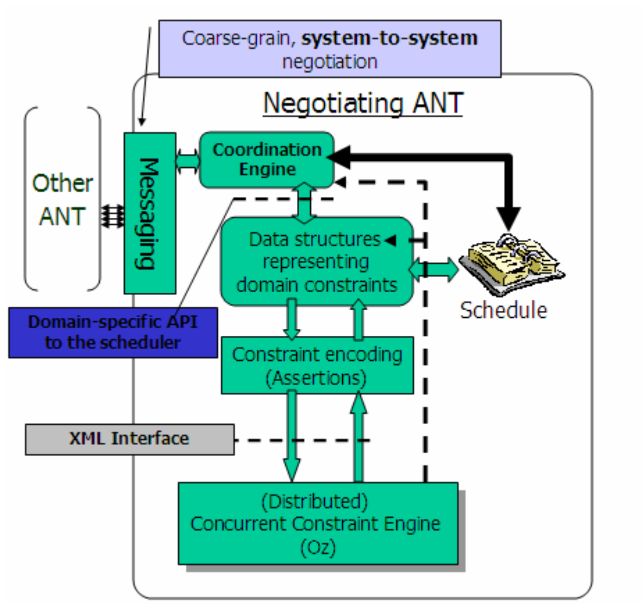


Figure 4: Layered architecture that permits translation of a problem into a set of constraints to be solved by a generic constraint solver.

Each constraint in Oz is implemented as a parallel thread, which is activated whenever variables under its control are changed. In this sense, the Oz engine implements a highly efficient negotiation process: each constraint represents a “concern” for negotiation, which gets evaluated upon changes to the entities that concern is related to. Furthermore, some of the constraints are implemented as “soft” constraints that can be violated during the search process, meaning that they can be dropped from the negotiation, without compromising the overall solution. These “soft” constraints are shown to the user at the end of the negotiation process as a series of warnings that list the constraints that have not been met and which were dropped during the negotiation process.

The key to constraint programming is the translation of the problem (in our case a scheduling problem) into a set of constraints that can be solved by constraint satisfaction algorithms such as the ones available in Oz. A substantial research and development effort has been dedicated to this process during the entire project. This is an evolutionary process. As the system complexity and demands from the end users increase, additional constraints need to be added. We have developed the software tools and architecture that permit incremental development without system overhaul. The layered architecture of MAPLANT is illustrated in Figure 4. The two main application specific components are the data structure module that is used to store and represent the constraints and the constraint encoding module that operates on the data structure and transforms it in assertions that can be processed by the Oz engines. At the time of writing MAPLANT deals with more than 10000 constraints!

3. Application of the technology to real-world problems.

The MAPLANT project followed a two-pronged approach: Development of new technological solutions and evaluation of these solutions by end users. Early in the project, contacts were established between the MAPLANT team and MAG 13 at YUMA. Engineers spent time on the base, learning the domain of application and eliciting needs from the users. Trips were made at regular interval to test, demonstrate, and gather feedback on the system under development. Figure 5 presents a time line that shows the major evaluation and demonstration events that occurred during the project as well as the increasing level of complexity of the tasks being addressed.

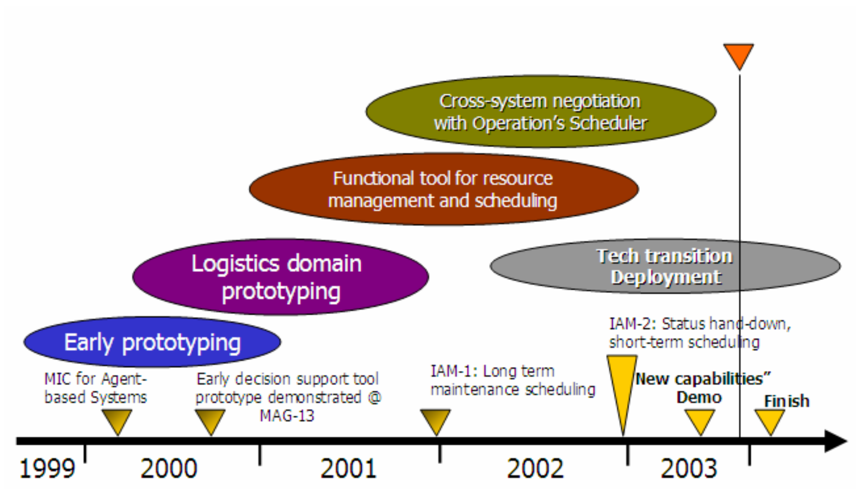


Figure 5: Important demo events and type of tasks being addressed.

The first year was spent adapting MIC technology to agent-based systems as well as learning the domain of application. In October of 2000 ISIS demonstrated the first prototype to MAG-13 personnel in Yuma, AZ. This was a decision support tool that used negotiating agents to aid aircraft maintainers in scheduling maintenance actions and it generated different options for actions depending on situations. At the time of IAM-1, the system was designed as an assistant to the MMCO for long term planning. Interfaces had been designed to permit the MMCO to assign airplanes to sorties if he so desired. The rest of the sorties were populated by MAPLANT, based on sortie description, equalization of airplane usage, and under- and over-utilization rules designed to overlap calendar and usage based inspections. A number of Graphical User Interfaces (GUIs) were designed to present the solution proposed by the system and permit the user to modify it. An acceptable solution was typically arrived at iteratively. A first solution was proposed by the system. Assignments made by the system were overruled by the user, which triggered another system assignment step, etc. The objective of IAM-2 was to support shift changes in the maintenance department of a squadron in addition to long term planning. At that time, MAPLANT was capable of generating the long term aircraft-to-sortie assignments, and to generate the lists of maintenance tasks required to support this assignment automatically. Shift-changes are the time at which the current status of maintenance tasks are reviewed and short term prioritization of tasks is made. This includes a review of currently open maintenance action forms (MAFs), a report by the various work centers on on-going tasks and unexpected delays, as well a status report on personnel/equipment available per work center. This review may result in changes to the planned schedule. For instance, a plane that had been assigned to a sortie cannot be made flight worthy before the mission starts because of a replacement part that has been ordered and that has not been delivered. This necessitates an airplane-to-sortie reassignment and reorganization and re-prioritizing of maintenance tasks. MAPLANT has been fitted with the GUIs that permit the end users to explore different solutions and study the impact of each of these on workload. A major advantage of MAPLANT over manual scheduling is that the long term impact of short term decisions is immediately apparent. For instance, reassigning a plane to another sortie may result in an overlap between long and difficult maintenance tasks on two aircraft a few weeks later. MAPLANT draws the attention of the MMCO on these problems and allows him to evaluate alternate solutions.

In addition to being fielded at YUMA, MAPLANT has been installed on an LHA and deployed with a Military Expeditionary Unit (MEU). Creating a shipboard version of MAPLANT has required implementing several extensions and improvements including (1) new utility tools for defining inspection (i.e. “task”) types that are not performed on land and including them in the schedule, (2) a tool for managing personnel databases, (3) improvements to the negotiation algorithms for addressing resource limits (number of available resources is different for different shifts), and (4) gathering user feedback and making continuous improvements to the system.

4. Technology Transition/Transfer

A substantial effort has been made to transition the technology and the system we have developed not only in the Marine Corps but also in other areas. We have worked with the CACE ACTD and USC/ISI on transitioning the tools to the Joint Strike Fighter (JSF) program. In early July 03 the Technology Transition Agreement with the JSF Program Office (PO) and Lockheed Martin (LM) JSF program has been signed. Together with USC/ISI, we have also attended the USAF Maintenance Integration Technology Working Group (MITWG) meeting in

Montgomery, AL, where requirements for maintenance scheduling tools were collected from USAF commands. ISIS has contributed the requirement for the joint operations/maintenance scheduling, which has been officially recognized. The USMC has created an office to tech transition CACE research results (including MAPLANT) to the service's aviation squadrons. The office will provide funding for facilitating the tech transition.

ISIS has also established contacts with the 45th Space Wing at Cape Canaveral, FL. This operation is charged with maintaining as well as improving and modernizing the space launch infrastructure and launch processing systems, while minimizing the impact system maintenance has on launch schedules or on the processing of flight hardware. To demonstrate the feasibility of transitioning the methodology, we have developed another application. ISIS chose to focus on providing planning and decision support tools for a single facility on the Cape--The Defense Satellite Communications System Processing Facility (DPF). The DPF is a USAF facility designed for off pad processing of spacecraft payloads. The facility's key features are its two high bay hazardous processing test cells (HPF) and a single low bay test cell (PPF). Each bay is a Class 100,000 clean room with a strictly controlled environment (e.g. typically 70±5°F with a relative humidity of 30 to 50%) and all of the support equipment necessary for the assembly, integration, fueling and processing of satellites, ordnance and solid rocket motors.

The application ISIS has selected is the planning and the scheduling of the maintenance tasks that need to be performed on the facility and the prototype we have developed has been named SpacePlant. Information about tasks to be performed, historical information about their duration and their detailed description, as well as the personnel and tools required to perform these are stored in various databases. The first task has been to transform the information contained in these databases into XML files that could be read by our system. This front-end component generates four XML files: the workload, the procedures, the manpower availability, and the tool files. These files are read by our system and used to create a schedule. In addition to these files, guidance (i.e., length of shifts) information provided via an additional XML file as is the case in MAPLANT. The same scheduling engine and the same GUIs and output files we have designed for aircraft maintenance have been used for this application, thus demonstrating the generic nature of the technology we have developed. In its current state, SpacePlant is capable of scheduling required maintenance tasks over a 2 weeks interval. Figure 6 shows a schedule generated by SpacePlant. Following current procedures, the schedule is separated into a 72 hours segment and an 11 days segment. When an unforeseen event happens, the schedule computed for the next 72 hours need to be kept as static as possible. Modifications in the schedule caused by the unforeseen events should be accommodated by modifying the long-term schedule. Figure 7 illustrate the GUI designed to show intervals over which scheduled maintenance tasks tax personnel resources. The horizontal position of the green vertical bars correspond to shift times. Their height corresponds to the available resources. The height of the pink boxes reflects needed resources. As can be seen in the example, personnel resources are almost completely utilized over 3 shifts. The blue shape is a rough estimation of required resources made prior to scheduling. If the scheduling engine fails to produce a solution due, for instance, to a lack of personnel resources, this information can be used to spot bottleneck areas and rearrange tasks to be performed or add resources temporarily.

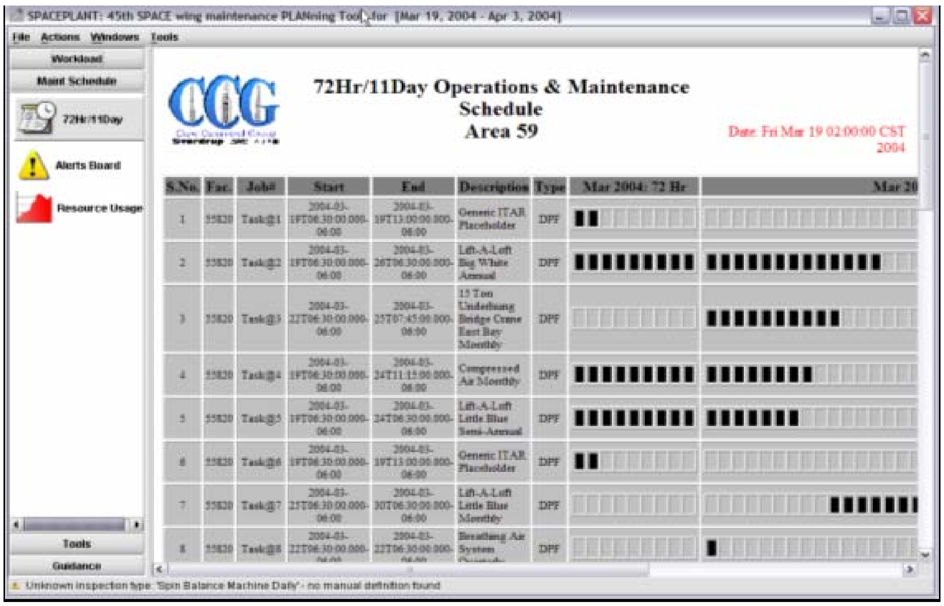


Figure 6: A schedule generated by SpacePlant

Figure 7: Risk Analysis in SpacePlant

5. Deliverables

5.1 Software

The most important deliverable is the MAPLANT system itself. It is available for download at the following URL:

www.isis.vanderbilt.edu/Projects/micants/maplant/index.html

This site not only provides the code, but also a complete on-line user's manual that describes the various components of the system, as well as a demo that walks the user through the various steps required to complete a schedule. SpacePlant can be downloaded from the same location.

5.2 List of Main Publications

van Buskirk C., Dawant B., Karsai G., Sprinkle J., Szokoli G., Suwanmongkol K., Currer,Russ: Computer-aided Aircraft Maintenance Scheduling, ISIS-02-303, November, 2002.

Sprinkle J., van Buskirk C., Karsai G.: Analysis and Representation of Clauses in Satisfiability of Constraints, ISIS-01-205, August 6, 2001.
Sprinkle J., van Buskirk C., Karsai G.: Modeling Agent Negotiation, IEEE International Conference on Systems, Man, and Cybernetics, Proceedings, Nashville, TN, October 8, 2000.

Karsai G., Bloor G., Doyle J.: Automating Human Based Negotiation Processes for Autonomic Logistics, Proceedings of the IEEE Aerospace 2000, CD-ROM Reference 11.0302, Big Sky, MT, March, 2000.

6. Conclusions and Discussion

MAPLANT is one of a set of tools developed under the DARPA-IXO ANTS that is part of the CACE ACTD suite of decision support tools, which has received very positive endorsement from end users. In May 2003, Col. Mark A. Savarese, USMC Commanding Officer, Marine Aircraft Group 13 stated "*During Combat ops I personally referred to the CACE suite of Decision Support tools three times a day….I wish all my units had CACE….the ones that did had the absolute highest readiness, and therefore combat capability…In a combat environment every minute of preparation is precious and considered a commodity….*"

The key to this success has been a tight interaction not only between scientists, engineers, and end users at each site institution involved in the project but also between institutions. An important lesson that has been learned is that a complete solution for a problem as complex as the one addressed in this work cannot be achieved at once. It is an evolutionary process that necessitates constant re-evaluation of directions, priorities, and goals. It is thus critical to develop the tools that permit rapid development without complete system overhaul. In our experience, constraint programming is an excellent approach for the type of problems we have dealt with. When a modification is required, the main difficulty is to translate new requirements into additional constraints. When this is done, the same generic engines can be used to find a solution, if one exists.

APPENDICES

Appendix 1: MAPLANT User's Manual. A printed copy of the on-line MAPLANT manual is included with this report.

Appendix 2: Summary PowerPoint slides that present the MAPLANT project. This presentation includes some of the figures included in this report.

Appendix 3: Summary PowerPoint slides that present the SPACEPLANT project. This presentation includes some of the figures included in this report.

Appendix 1

MAPLANT User's Manual. A printed copy of the on-line MAPLANT manual is included with this report.

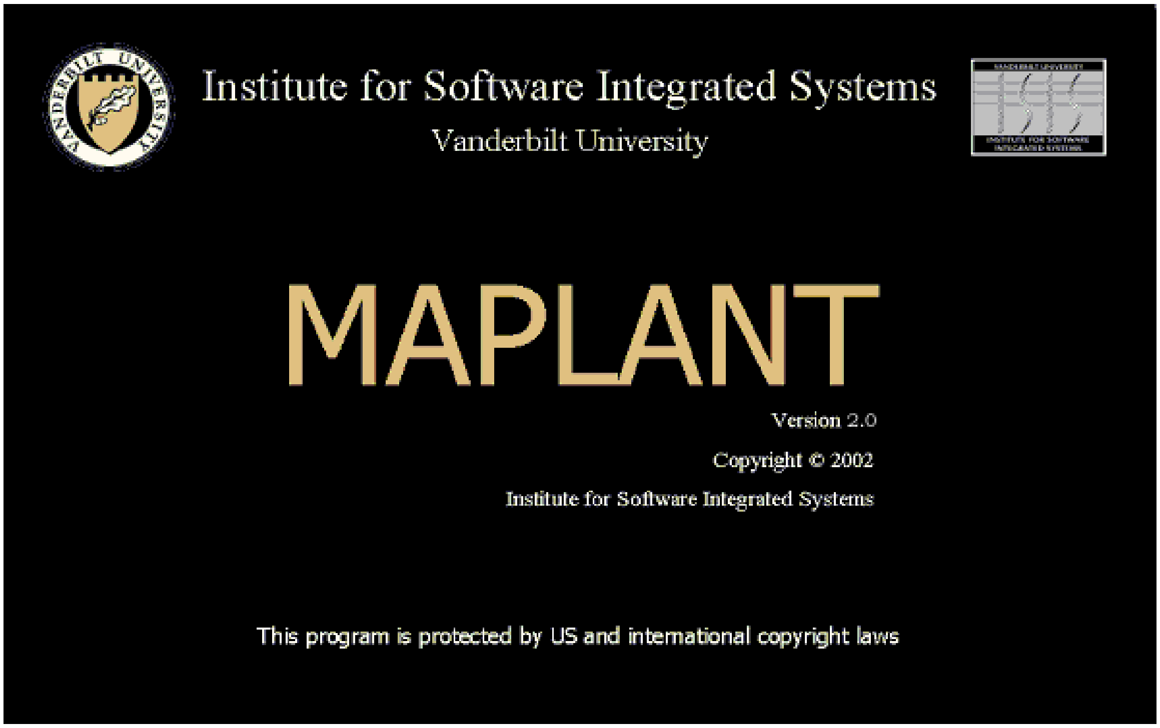
Maplant User's Manual

Table of Contents

MAPLANT
The MAPLANT User's Guide

June 2004

Last modified on June 29th, 2004 8:05 PM



Submit support requests to maplant-support@isis.vanderbilt.edu

Getting Started

Concept of Operations

The Coherent Analytical Computing Environment

MAPLANT, the software application documented in this manual, is but one of several programs available in the CACE suite of tools. This section of the manual describes the vision of the CACE project as a whole, and it shows where MAPLANT fits into this *bigger than the sum of its parts* system of systems.

CACE is designed to be a command level decision support intra-system. The tool suite consists of a set of modular tools designed to create an analytic, shared data environment integrated across functional domains. CACE is linked laterally and vertically by networks of proactive intelligent applications and agents, enabling reduction in task loading as well as facilitating improved decision making, safety and efficiency at all operational levels.

The CACE Software

CACE is a collection of integrated software tools designed to address several aspects of marine aviation. The primary software components comprising the CACE suite of tools is as follows:

- **Data Warehouse** - The data warehouse provides three functions within CACE. Firstly, it is used to maintain historical NALCOMIS data so that the maintenance history of a unit over long periods of time is saved. This, for example, supports the operation of trend analysis. Secondly, the warehouse is used to automate the creation of certain NAMP reports, such as the daily AMRR. Individual users may even define their own customized report formats. Finally, the warehouse serves as an interface between CACE and NALCOMIS. The warehouse is the only CACE application authorized to connect to the NALCOMIS host.

- **MAPLANT** (Maintenance Planning Tools) - MAPLANT, the primary subject of this user's manual is but one of many software tools provided by CACE. The main function of MAPLANT is to assist maintenance control with (a) planning for and accomplishing long-term maintenance goals and (b) managing maintenance operations in support of a flyday. MAPLANT is designed to play the role of a decision support tool that helps to point out risks and to suggest alternative ways of mitigating these risks.

- **MSR** (Mission Sensitive Resumes) - MSR provides many services to the community of CACE software and human agents. One example is The Electronic Aircraft Descripancy Book (eADB). The eADB facilitates in-depth study of an aircraft's personality and maintenance history. As opposed to the current practice of maintaining and reviewing the history of the only the last 10 flights, the eADB has access to maintenance records for much longer periods of time. It is even possible, for example, to review only the maintenance history of a jet that is relevant to a specific mission profile. This ability to pinpoint mission-specific maintenance issues or to recommend bunos for particular types of missions is sometimes referred to as **MSAR** or mission sensitive aircraft resumes. A key design goal of the eADB is to, from massive quantities of data, filter out only the details that are relevant or interesting for a specific user and a particular mode of operation.

- **SNAP** (Schedules Negotiated by ANT Planners) - SNAP is used by operations to derive flight plans, which support the goals and intentions of the operations department. Since SNAP and MAPLANT may be connected together, each can be used to analyze the impact of the sister department's plans upon local operations.
-

The CACE Hardware

In the Concept of Operations, all CACE laptops will be connected to the unit's LAN and will exchange files electronically. File/data exchange may happen automatically or as directed by the user. At present, the functional nodes are as follows:

The **Maintenance Control node** will function as the warehouse server to the NALCOMIS host. It will maintain the warehouse database and produce the current list of NAMP reports. In addition, the maintenance control node will provide data to resident MAPLANT and to MSR applications. It will provide MAPLANT data files to SNAP via a shared drive on the LAN (or optionally via email or sneaker net), and It will provide data to the MSR application at both the Schedule writer and Operations Duty Officer nodes. The maintenance control node will provide for prioritization of workcenter goals at shift change, monthly planning of scheduled, calendar and usage-based maintenance, allow assessment of daily and monthly schedule supportability as well as impact on the monthly plan. MSR will assist in rapid aircraft MAF review, trend analysis by the system, and filtering of mission relevant information.

The **Ops Schedule Writer node** will function as the primary daily, weekly and long-range flight scheduling and planning tool. It will draw SARA data from the designated SARA database location (SARA laptop or shared drive on the LAN), data files from MAPLANT, and produce certifiable daily schedules for use by the ODO in execution. The Ops Schedule Writer node will send necessary planning data to MAPLANT via the shared LAN drive.

The **ODO node** will be responsible for capturing (via ODO manual entry) actual schedule execution details and any changes. After certification, it will pass update information for the SARA database update to the OPS Clerk SARA node and notify the Schedule Writer node of changes. In addition the ODO node will have MSR functionality for ODO/Pilot pre/post briefing ready room review of assigned aircraft.

The **Ops Clerk node** (SARA only laptop), will receive updates and use the information as normal to correct data, manage log books, etc.

The CACE Users

At the present time, the primary user roles supported by CACE are: the ODO, pilots, the ops clerk, flight schedule writers, maintenance admin analysts, the AAMO, the MMCO and maintenance controllers. MAPLANT primarily supports the MMCO, maintenance controllers and ODOs.

Software Installation

Installing MAPLANT

To install MAPLANT, please execute the *maplant-setup.exe* program on your distribution media. If you do not have an installation CD, direct inquiries to maplant-support@isis.vanderbilt.edu. The installation program is a standard InstallShield™ executable, which requires Microsoft Windows™. The installation may require a password; again this password may be obtained from the above MAPLANT support email address.

As explained in the *Readme Information* presented during the installation procedure, MAPLANT requires separate installations of some third-party software (e.g. Microsoft Project 2000).

NOTE: The MAPLANT setup process does not install other CACE tools such as SNAP. If you choose an installation directory other than the default--C:\Program Files\CACE--please install any other CACE tools in that same location. For example, if you prefer to install on the D: drive due to space constraints, please install SNAP and the eADB in the CACE subdirectory of your D: drive.

Upgrading MAPLANT

If you have a previous version of MAPLANT already installed and it has been in a production environment requiring maintenance of data such as the definition of which mechanics are assigned to which shifts, your squadron's data must either be reported to ISIS before the release of a new version, or it must manually be imported after upgrading.

By default, data such as this **will be lost** when upgrading by running the setup program for a newer version. Please contact maplant-support@isis.vanderbilt.edu for technical support in managing this upgrade process.

If your MAPLANT is not an active, production system (e.g. just a demonstration prototype), this issue does not concern you. You should however, always uninstall any previous versions of MAPLANT before installing a newer version (i.e. using MS-Windows' *Add or Remove Programs* utility).

Starting MAPLANT

After successfully installing MAPLANT, you should have a CACE/MAPLANT folder of icons in your Programs folder (start from the *Start* button on bottom left corner of your desktop). The wrench icon is used to execute the maintenance command advisor that is the central interface to the MAPLANT tools.

Startup Screens

After clicking the wrench icon to start MAPLANT, you should first see a splash screen, and optionally some dialog boxes and informational messages as MAPLANT consumes data from the warehouse and analyzes the maintenance status of the unit.

Figure 1. The Splash Screen

MAPLANT will infrequently provide informational message to the user by way of a green-on-black textual log window. The following example shows that MAPLANT is complaining about inspections currently listed in NALCOMIS that it doesn't understand how to perform.

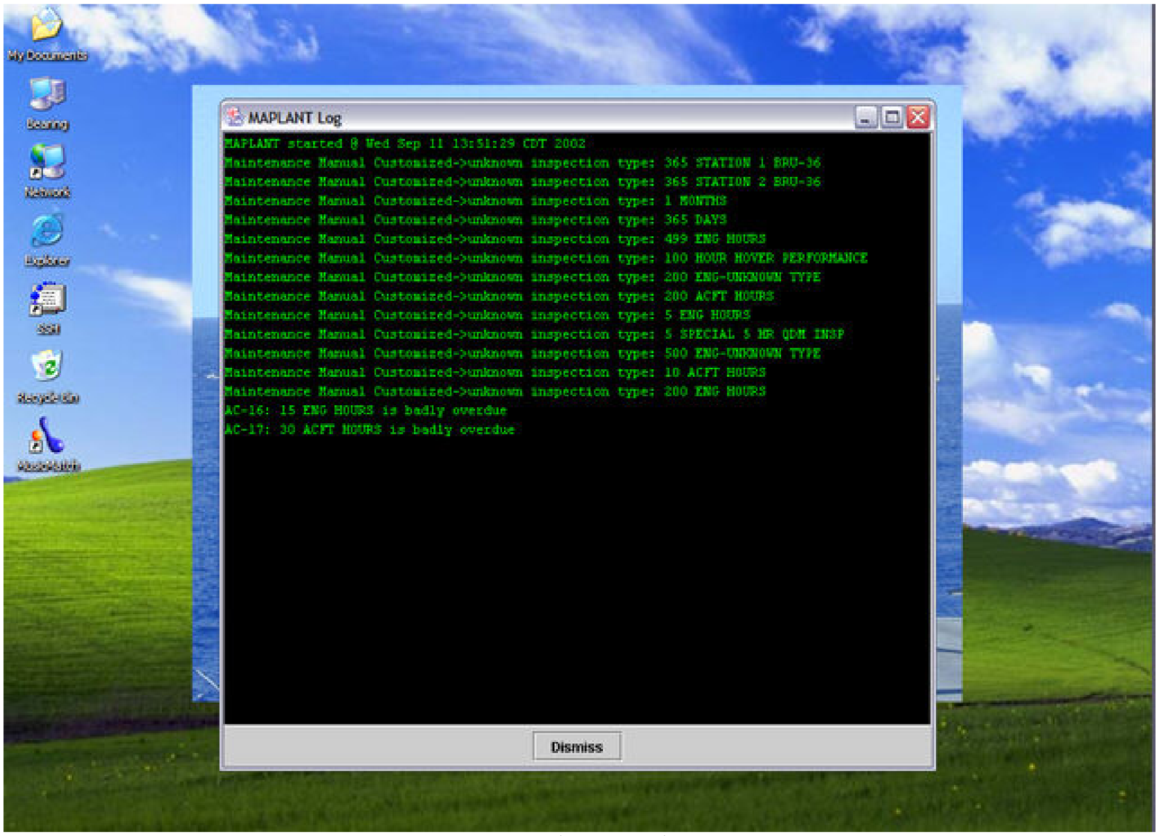


Figure 2. The Log Window

Note that when the SOPs (Standard Operating Procedures) facility of MAPLANT is introduced later in this tutorial, we'll show how to remedy this type of error by supplementing the standardized Naval maintenance manuals with locally defined maintenance procedures or by informing MAPLANT of typing/spelling errors that haven't yet been purged from NALCOMIS.

The following figure shows another type of error condition that you may encounter during startup. MAPLANT has identified a scheduled inspection that should *legally* have already been signed off, but that is not yet completed according to NALCOMIS. This could, for example, be due to the lag between the actual maintenance actions and the official recording of them in computer systems.

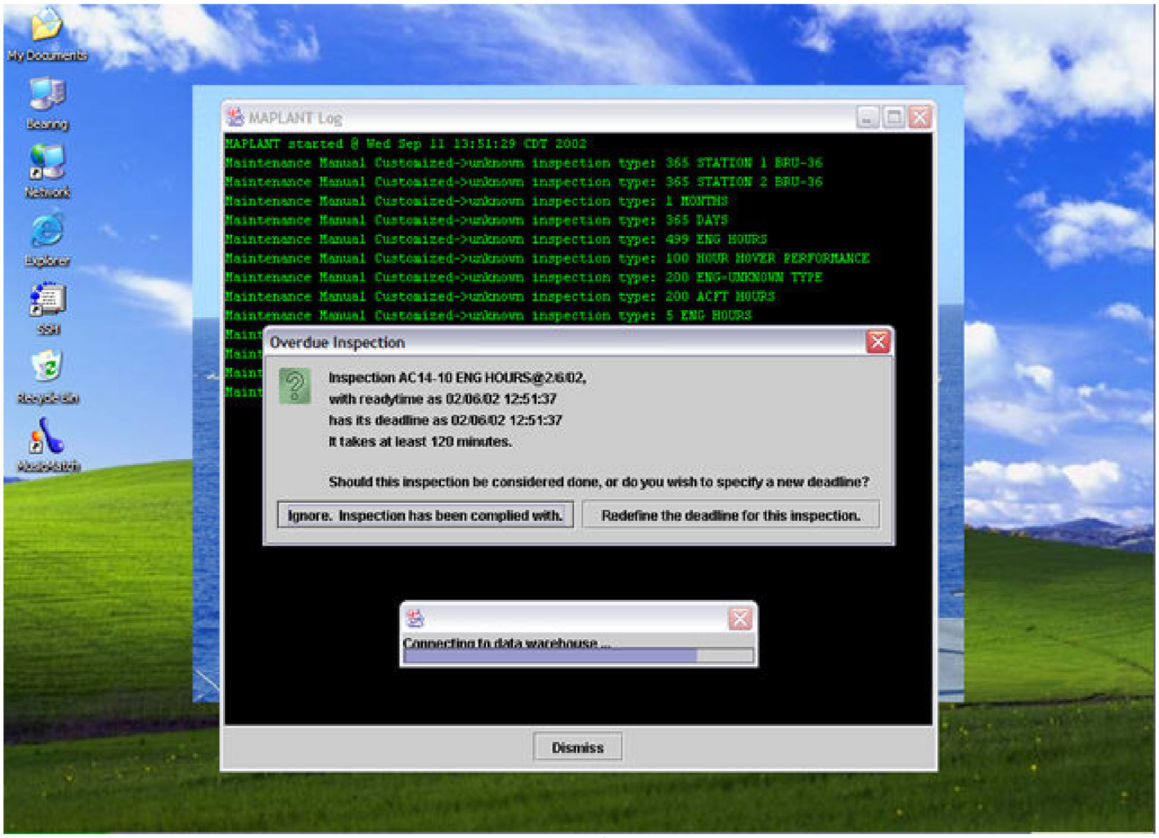


Figure 3. An Overdue Inspection

Finally, another common startup dialog is presented in response to MAPLANT having guidance about a specific aircraft that, according to NALCOMIS, does not belong to the squadron. This could, for example, be due to a recent aircraft transfer.

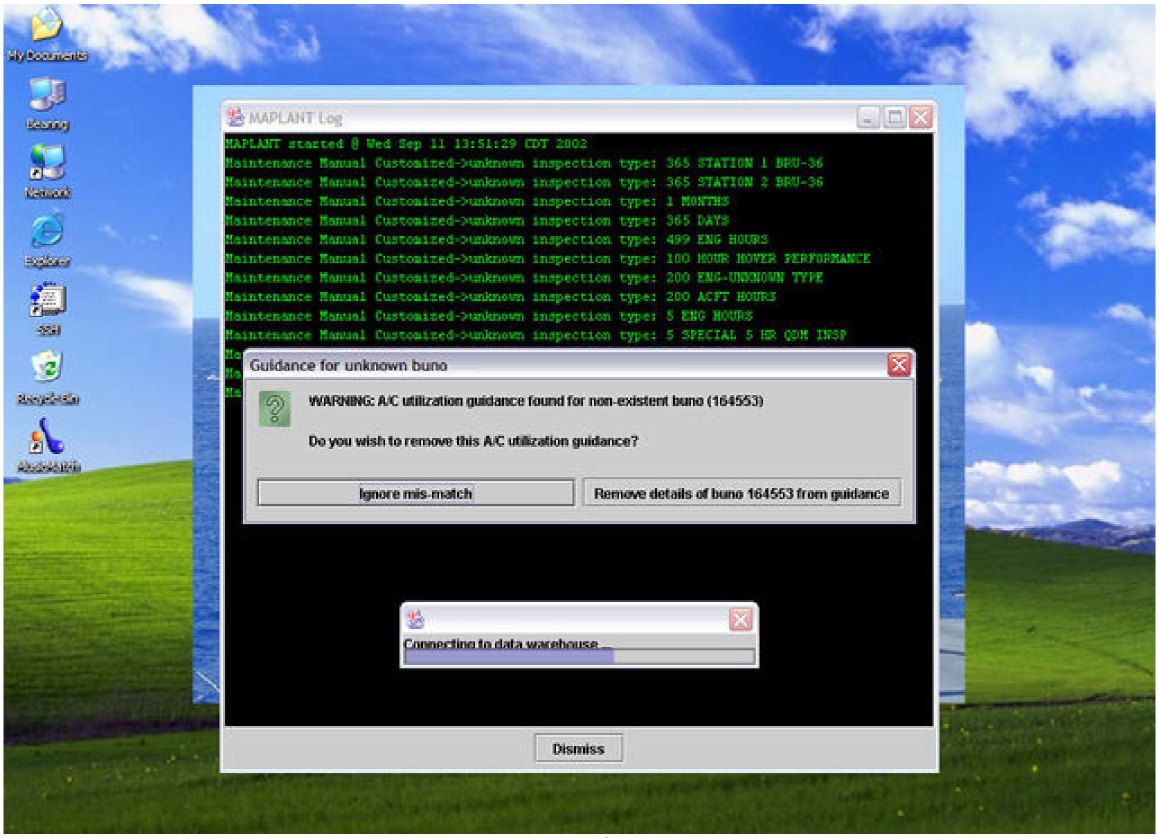
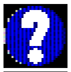


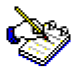
Figure 4. Transfer Dialog

Other MAPLANT Tools and Icons

The other utilities provided by MAPLANT are discussed in later sections of the manual. They are as follows:


The on-line help system


The MAPLANT *Readme* file


Customize your configuration

Edit the Standard Operating Procedures


License the product

Maintain the personnel information


Maintain the support equipment inventory


Select or merge flight schedules.


Maintain the repair manuals.

Introductory Screens

Aircraft Status Board

The *A/C Status* screen appears in the main window at system startup. It reflects the aircraft status report the system believes is accurate from information communicated to MAPLANT by the Data Warehouse. The time stamp, shown in the upper left corner, indicates the time of the warehouse's last download from NALCOMIS. The status report shown below is from 05:31 on Nov 29th (day 333 of the year 2001).

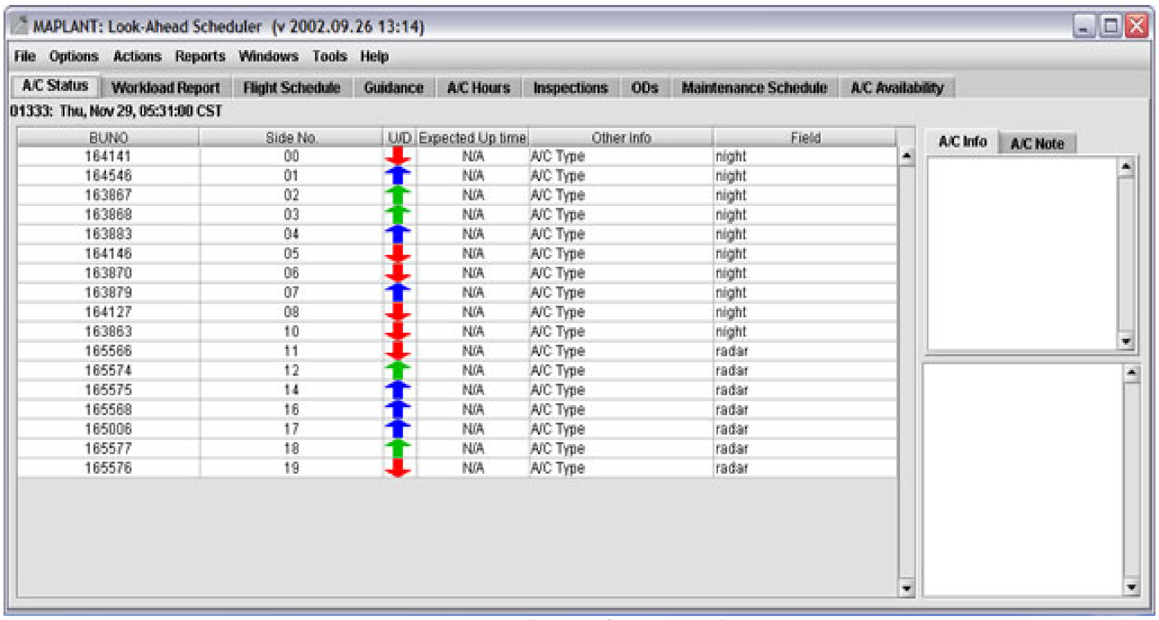


Figure 5. The Aircraft Status Board

Each row in this table represents a single aircraft. The individual columns show particular attributes such as bureau number or side number for each jet. The Up/Down icons have the following meaning:

- Full Mission Capable
- Partial Mission Capable
- Non Mission Capable
- Time-Till-Walk (see below)
- Expected Up

If an aircraft is NMC, you may estimate the time at which it will become operational by clicking in the appropriate *expected up* cell (for example, if you have knowledge about parts availability that isn't in NALCOMIS). A dialog window will pop up, allowing you to specify the year, month, day and time that the plane is expected up. Press the *Clear* button on this dialog, to remove an expected up time.

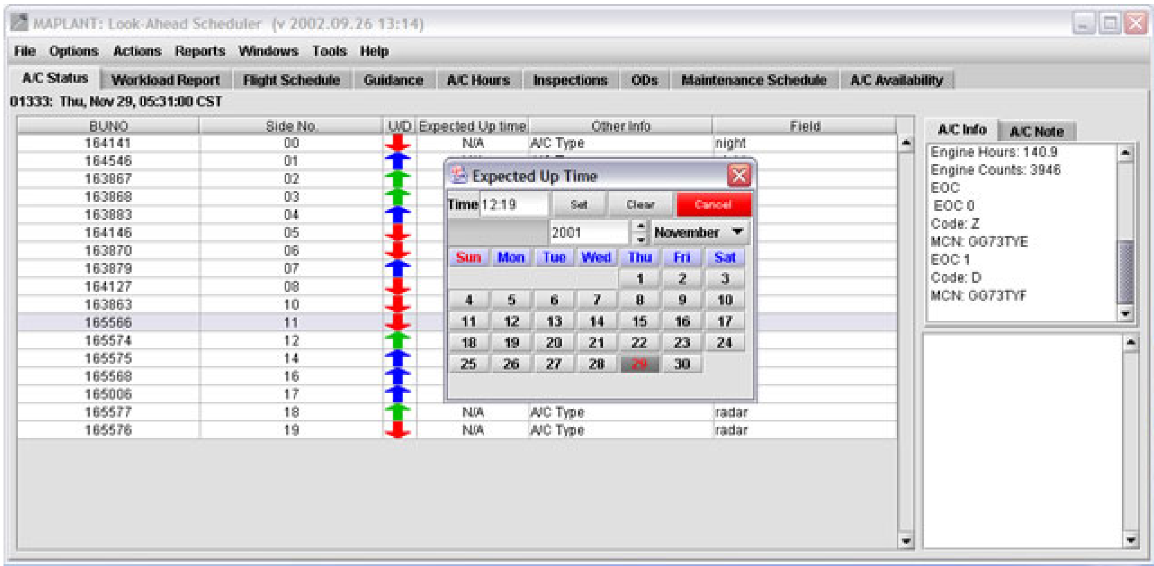


Figure 6. Set/Clear expected up time

Similarly, by **right-clicking** on a green or blue arrow, you can set the *time-till-walk*. This will cause the icon to change to a sideways arrow, indicating that the plane is almost ready for flight. After estimating when the final turnaround inspection will be completed, the MAPLANT node in the pilots' ready room will count down the time until pilots should walk to take possession of the jet. The count down timer will change to a red font and start counting backwards (i.e. minutes delayed) if the plane is not set back to F/PMC by the estimated walk time.

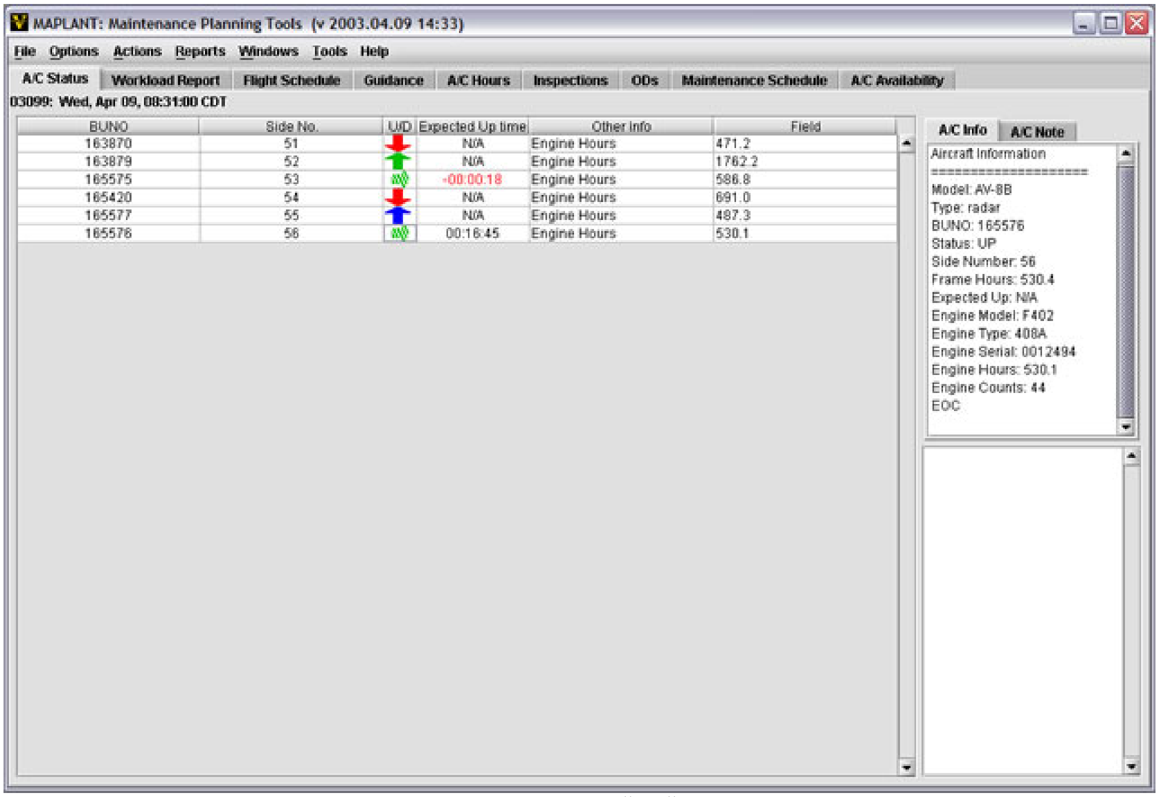


Figure 7. Time-Till-Walk

Finally, the *Field* column in the table displays values for the attribute chosen in the *Other Info* column. *Other Info* is a drop-down menu that may be used to show particular attributes. The default attribute is engine hours. Other examples are engine serial number or flight hours accumulated on the airframe.

To the right of the table area is three more gui widgets: (a) *A/C Info* (b) *A/C Note*s (c) and a general notes section. The aircraft info lists all information for the aircraft or row currently selected in the table. The two notes sections may be used as a scratch pad to document status not maintained in NALCOMIS (e.g. passdown information). The contents of *A/C Note* shows any notes specific to the currently selected aircraft row, while the contents of the notes section in the bottom right corner of the screen does not depend on the currently selected aircraft.

Workload Report

The *Workload Report* screen lists all known outstanding unscheduled maintenance. By default, unscheduled maintenance is not placed on the agenda by MAPLANT. Individual work items may be re-prioritized, however, so that MAPLANT will ensure that these items are considered when computing the maintenance schedule. This may be done manually, on a case by case basis, or automatically as the result of assigning an air vehicle to a sortie that it is not currently capable of flying (without corresponding repairs). The priority levels are REQUIRED, HIGHLY-DESIRABLE, DESIRABLE, NICE-TO-HAVE, DONT-CARE, and PROHIBITED. The color coding below is determined by the assigned priority of each job. Bright red represents maintenance actions that are REQUIRED to be successfully scheduled before the plan is considered legal and valid.

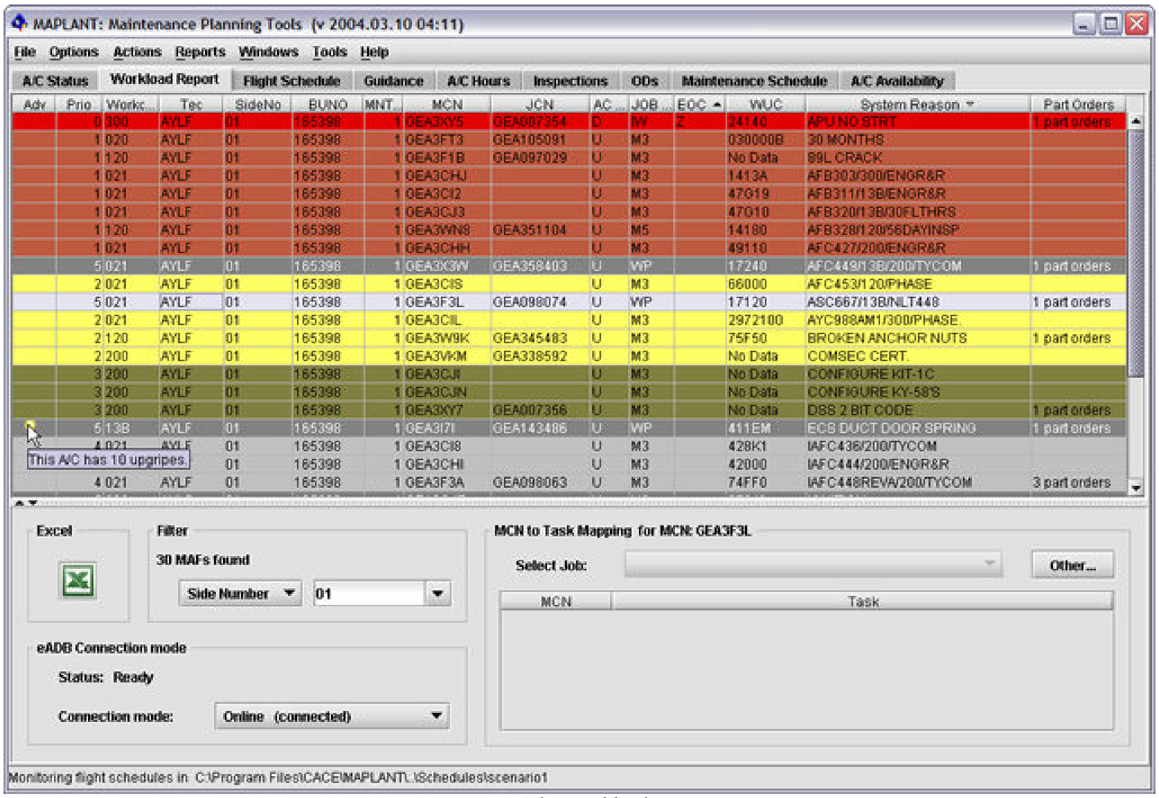


Figure 8. The Workload Report

The *Filter* section of the interface allows viewing specified subsets of the data. By selecting a column header in the table, you specify a primary key for the table rows to be sorted on. The example above shows all MAFs sorted in descending order based on their *EOC* code. Also, in the screen shot above, a secondary sort key based on *System Reason* has been defined. The secondary sort key is selected by holding down the SHIFT key while left-clicking on a column header. The *Priority*, *Workcenter*, *Job Status*, *WUC*, and *Part Order* table cells have descriptive mouseovers that provide more detail when the mouse pointer is allowed to hover over them. Finally, note the *Advice* light bulb in the example above, which warns of an aircraft that is breaking the *"no more than 10 up-gripes"* preference for not letting many, many, small and relatively minor items accumulate on any one vehicle. This report may also be exported to Microsoft Excel.

The *MCN to Task Mappings* and the *eADB Connection mode* sections of this interface will be discussed later in the advanced sections of the manual concerning the scheduling of repairs.

Flight Schedule

Clicking on the *Flight Schedule* tab just underneath the menu bar provides views of the individual flydays under consideration. The label in the top left corner above the table shows the dates covered by this flight schedule, as well as the currently selected day. The table shows the individual events and the calendar widget in the upper right corner is used to move from flyday to flyday. Be sure to change the year when switching from December 31st to January 1st.

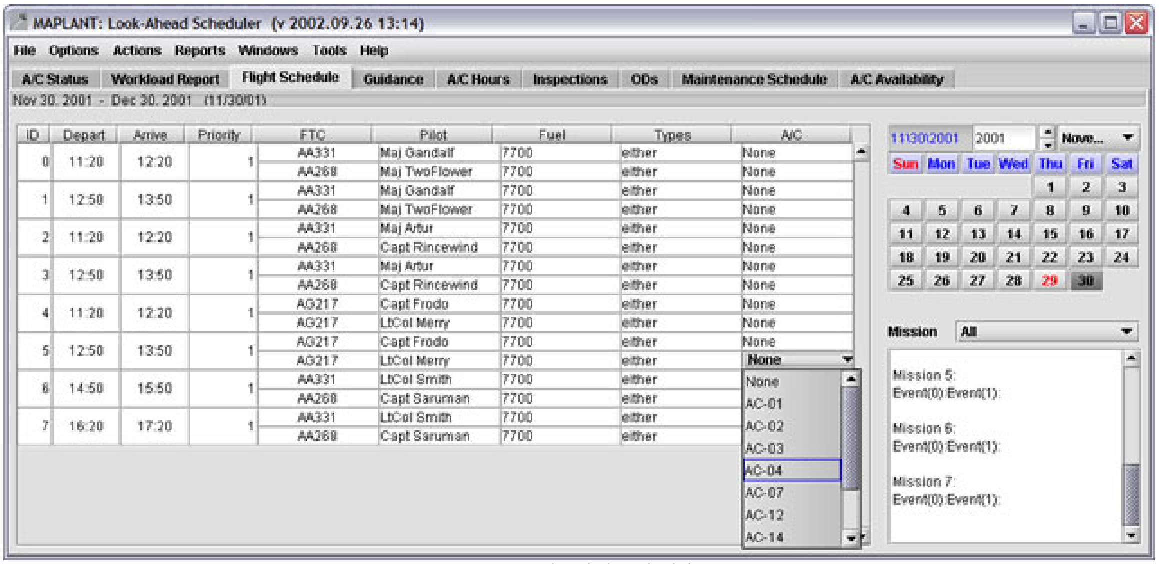


Figure 9. The Flight Schedule

The right-most column in the table shows the primary aircraft that has been assigned for each sortie. Note that this screenshot depicts the beginning of a planning process, where the user has not made any assignment decisions yet. Clicking in an a/c cell provides a prioritized list of aircraft capable of flying a specified mission. The font used to represent assignments and potential assignments has the following meaning:

AC-00	This assignment is recommended.
AC-01	This assignment would result in switching planes during a pit-event.
AC-02	While this aircraft is legally capable of fulfilling the role, the assignment would conflict with guidance defined by the user (more details about this will follow in later sections).
AC-03	Certain non-functional subsystems on the aircraft are currently affecting the jet's ability to fulfill this mission. Making this assignment will result in *missionized maintenance* that must be completed before the departure time of the event.
AC-	Assignments rendered in boldface indicate that a human made this

04 choice. We will see later that it is possible to instruct MAPLANT to make intelligent assignments that conform to overall guidance used to steer the decision process. MAPLANT, however, will never override an assignment chosen by the operator.

Guidance

The next set of screens allow the human maintenance expert to define some high-level directives, which MAPLANT uses as general rules to help guide it when looking for solutions to maintenance dilemmas. There are five such screens under the *Guidance* tab:

1. Scheduler Options
2. Maintainers
3. Spares
4. Rules &
5. Aircraft Utilization

The aircraft utilization guidance gui is shown below. This is where the user instructs MAPLANT about how each of the aircraft should be flown throughout the planning period. The default policy is to equalize flight hours across the fleet, but this default behavior may be overridden by providing specific guidance for a particular jet.

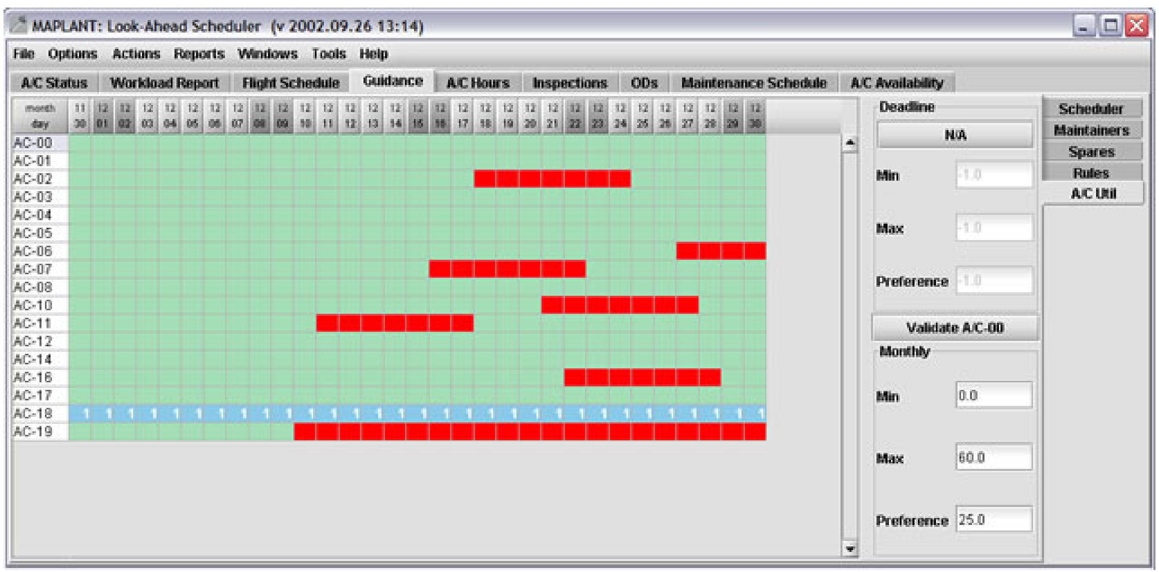


Figure 10. Aircraft Utilization Guidance

In the example above, MAPLANT has been instructed not to fly AC-18 for more than one flight hour per day, and AC-19 has been grounded beginning on the 10th day of the month. While the two entries for AC-18 and AC-19 were manually entered by an operator, at initialization time MAPLANT also automatically generates some utilization guidance based on what is known about upcoming scheduled inspections. The six smaller red stripes above are examples of utilization guidance generated by the system. In the above case, each represents a due window for upcoming 56-dsi inspections. By default, MAPLANT will *prefer* not to fly a plane during the windows within which a 56-dsi-s may be done. The plane will be used as a spare only, unless this default guidance is overridden. Depending on the operations tempo, however, a maintenance controller will sometimes be required to provide more specific guidance than this generic default, in order to fly all of the events in the schedule with the available aircraft. Specific examples of this, as well as explanations of the *deadline guidance* and the *monthly guidance* sections of the utilization screen will be presented in upcoming demonstration scenarios.

Another useful guidance screen is accessed by selecting the *Maintainers* tab (along the right edge of the window). This is where shifts, holidays and safety margins for resource consumption are defined.

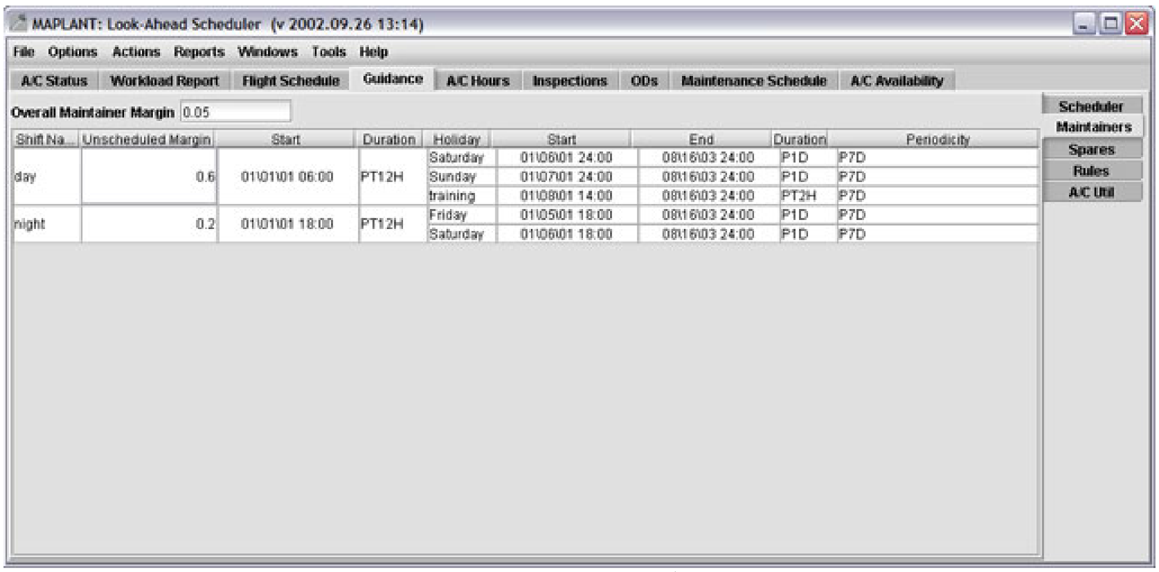
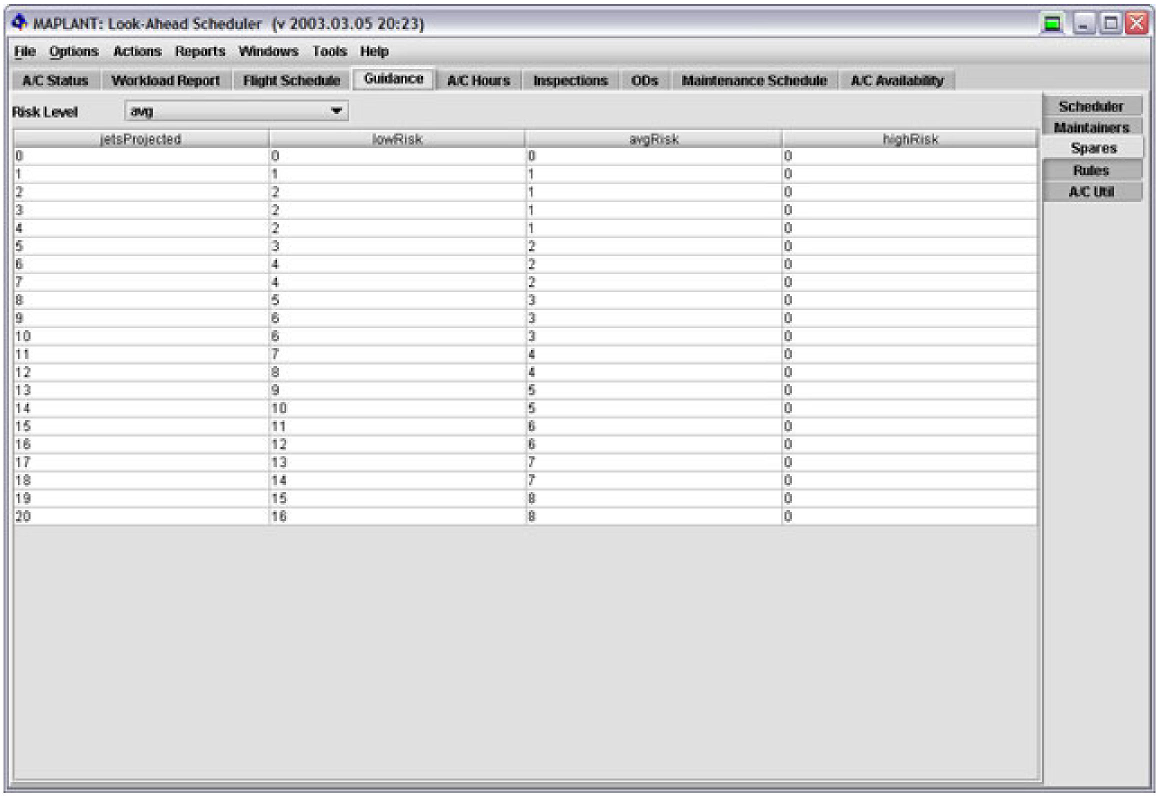


Figure 11. Maintainers Guidance

Above, we see that two shifts have been defined. The first is called "day", and it is a 12 hour shift where Saturdays, Sundays and a two hour block on Mondays are not devoted to aircraft inspections and repairs. The night shift is also twelve hours, but this group of workers is typically off on Friday and Saturday nights. Additionally, each shift has an *Unscheduled Margin* that is used to reserve some amount of resource capacity to deal with unexpected events. In the above example, we have instructed MAPLANT that when creating maintenance schedules, it should always attempt to keep (a) 60% of the day shift and (b) 20% of the night shift available to respond to unforeseen emergencies. Finally, there is an overall safety margin of 5% applied to all resources.



jetsProjected	lowRisk	avgRisk	highRisk
0	0	0	0
1	1	1	0
2	2	1	0
3	2	1	0
4	2	1	0
5	3	2	0
6	4	2	0
7	4	2	0
8	5	3	0
9	6	3	0
10	6	3	0
11	7	4	0
12	8	4	0
13	9	5	0
14	10	5	0
15	11	6	0
16	12	6	0
17	13	7	0
18	14	7	0
19	15	8	0
20	16	8	0

Figure 12. Spares Policy

The spares policy window allows you to define up to three different strategies for managing backup aircraft in the event that the originally scheduled plane is unable to fulfill its commitment for a particular sortie. Using the *risk-level* widget at the top left of the display, one may switch between *low-risk*, *average-risk* and *high-risk* modes of operation. The low-risk policy provides the most spares and is therefore least likely to result in a plan that is susceptible to failure. Each of the three modes defines the quantities of spares desired for a specific situation. The left-most column specifies the tempo that is being considered in each row. For example, the fourth row above states that *if three jets are required at some point in the flight schedule, maintenance should plan to provide (a) 2 (low-risk) (b) 1 (avg risk) or (c) 0 (high-risk) spares at that point in the schedule.* The individual values in the in the three right-most columns are configurable.

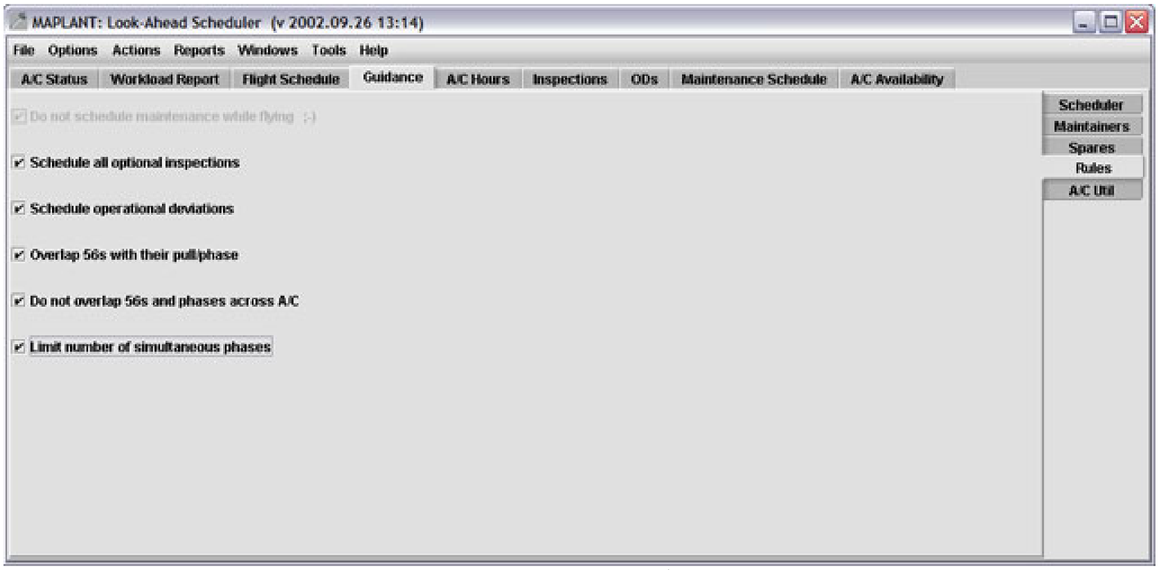


Figure 13. Maintenance Control Strategies

Finally, the last example guidance screen presented in this section shows where various modes of operation may be turned on or off before computing a maintenance schedule. For example, above we've instructed MAPLANT to, whenever possible, not schedule a phase inspection while a 56-dsi is being performed on some other aircraft (i.e. the *Do not overlap 56s and phases across A/C* rule). If this strategy is turned on and MAPLANT is unable to determine a solution that conforms to the guidance, you will be informed of the specific violations at the time of schedule generation. At that point, it may be possible to reframe the problem in order to make this possible (by tweaking various other pieces of guidance), or you may simply have to accept that the plan isn't ideal (and be on the lookout for the side effects of the specific guidance violation during plan execution).

Aircraft Hours

The next two screens discussed are used to (a) highlight problem flydays in the proposed flight schedule and (b) analyze the projected utilization of aircraft during the execution of this plan. The screenshot below shows, by day, the number of flight hours planned by ops and to what level of confidence the maintenance plan supports these flights. Blue indicates that there is an excess of flyable jets during these periods. Green indicates that the number of sortie assignments plus spare jets is optimal, according to the spares policy defined under guidance. Yellow indicates that, while aircraft are scheduled for the proposed flights, the number of spare aircraft is less than that desirable. Finally, red indicates sorties that are not supportable according to the current maintenance plan.

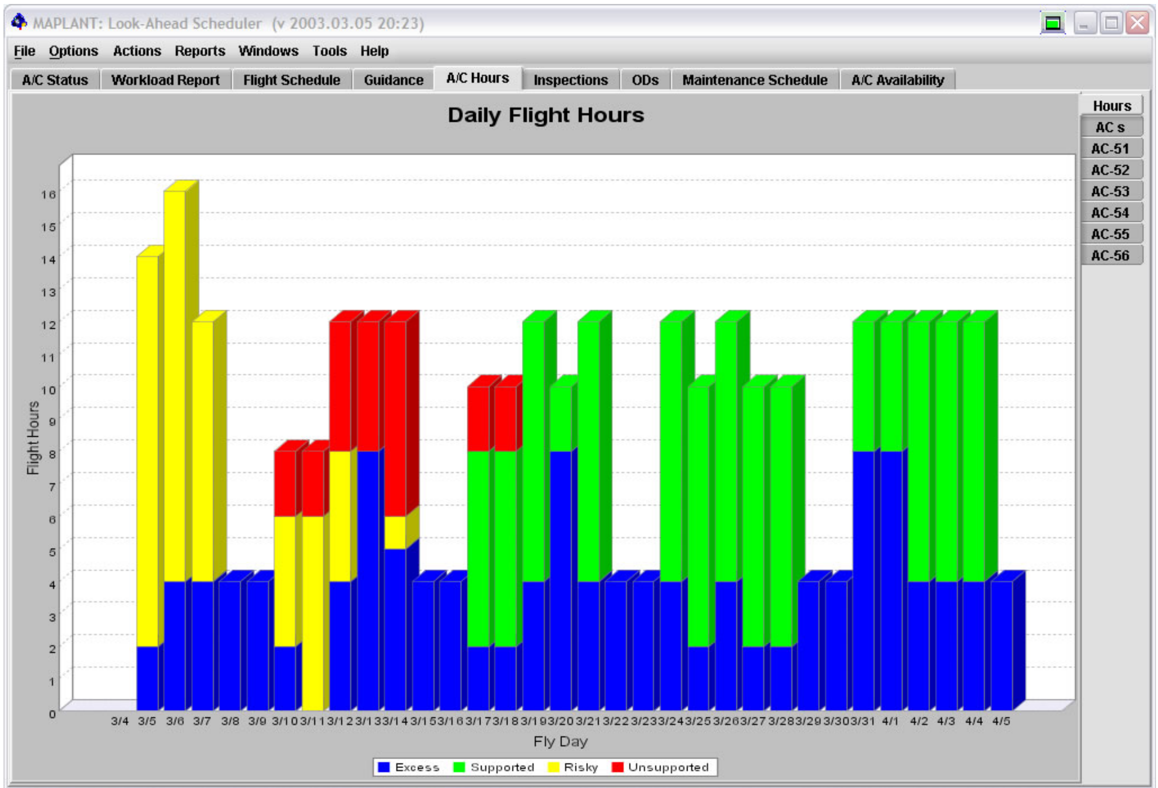


Figure 14. Proposed Flight Hours By Day

Frequently, MAPLANT is unable to automatically find an allocation of aircraft to sorties that conforms to all given guidance. Above, we see that there are approximately twenty two events that MAPLANT was unable to satisfy during the first iteration of the planning session (shown as red bars). At this point, it would be up to the controller to help resolve the conflict between the high-level maintenance guidance and the operational tempo specified in the proposed flight schedule. In some cases, this is possible by providing more specific guidance directed at these problem events. In other cases, certain events must be sacrificed, for the time being. In this later situation, at the end of the maintenance planning process, MAPLANT could contact the SNAP scheduling system about these problem sorties, and ops has the option of either deleting the events, or reiterating the ops plan in order to take advantage of more detailed information about the flight schedule's impact on the maintenance department (now available from MAPLANT). For example, MAPLANT might suggest moving an event to an area in time where fewer 56-dsi-s are planned, or the sortie could be placed into a pit event, effectively reducing the number of aircraft required to support the flight schedule.

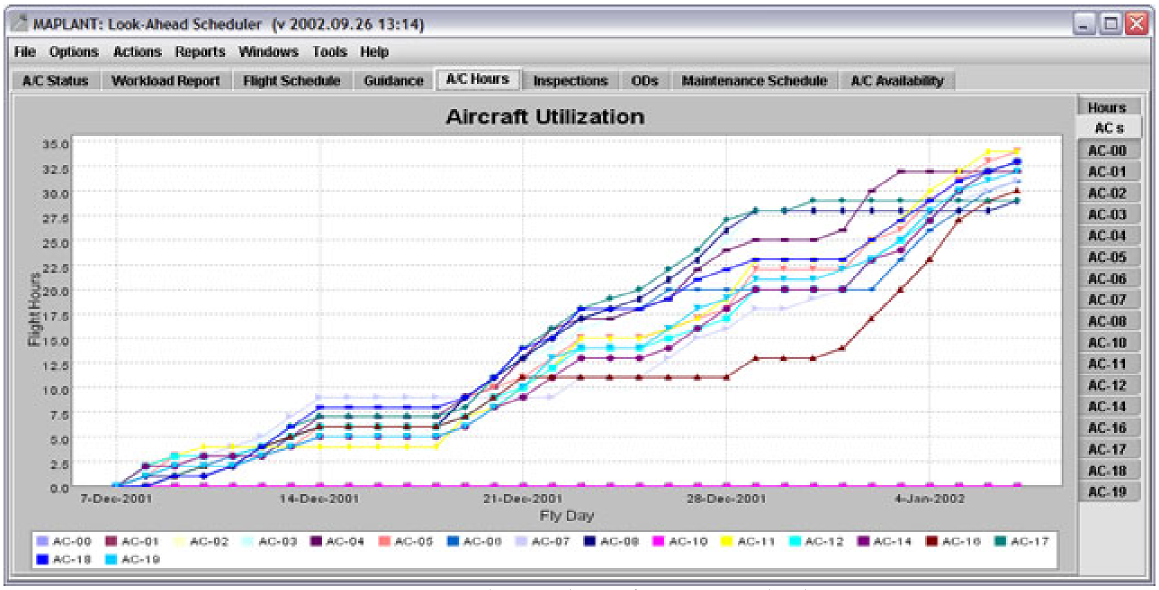


Figure 15. Projected Accumulation of Hours Across the Fleet

Finally, the *ACs* tab gives an overview of how the jets are expected to accumulate flight hours during the planning horizon. These plots are used as a quick sanity check to ensure that no plane is being over or underflown (or if it is, then this is due to some guidance that was entered by the operator). For example, any curve that rises too steeply may indicate a potential for breaking the plane during that period due to overuse. Similarly, an aircraft that sits idle for a long period may start to degrade.

Upcoming Inspections

The *Inspections* screens give various views of the scheduled inspections that will fall within the planning horizon. For example, the following figure shows an example report that could be pasted into next month's Monthly Maintenance Plan (MMP).

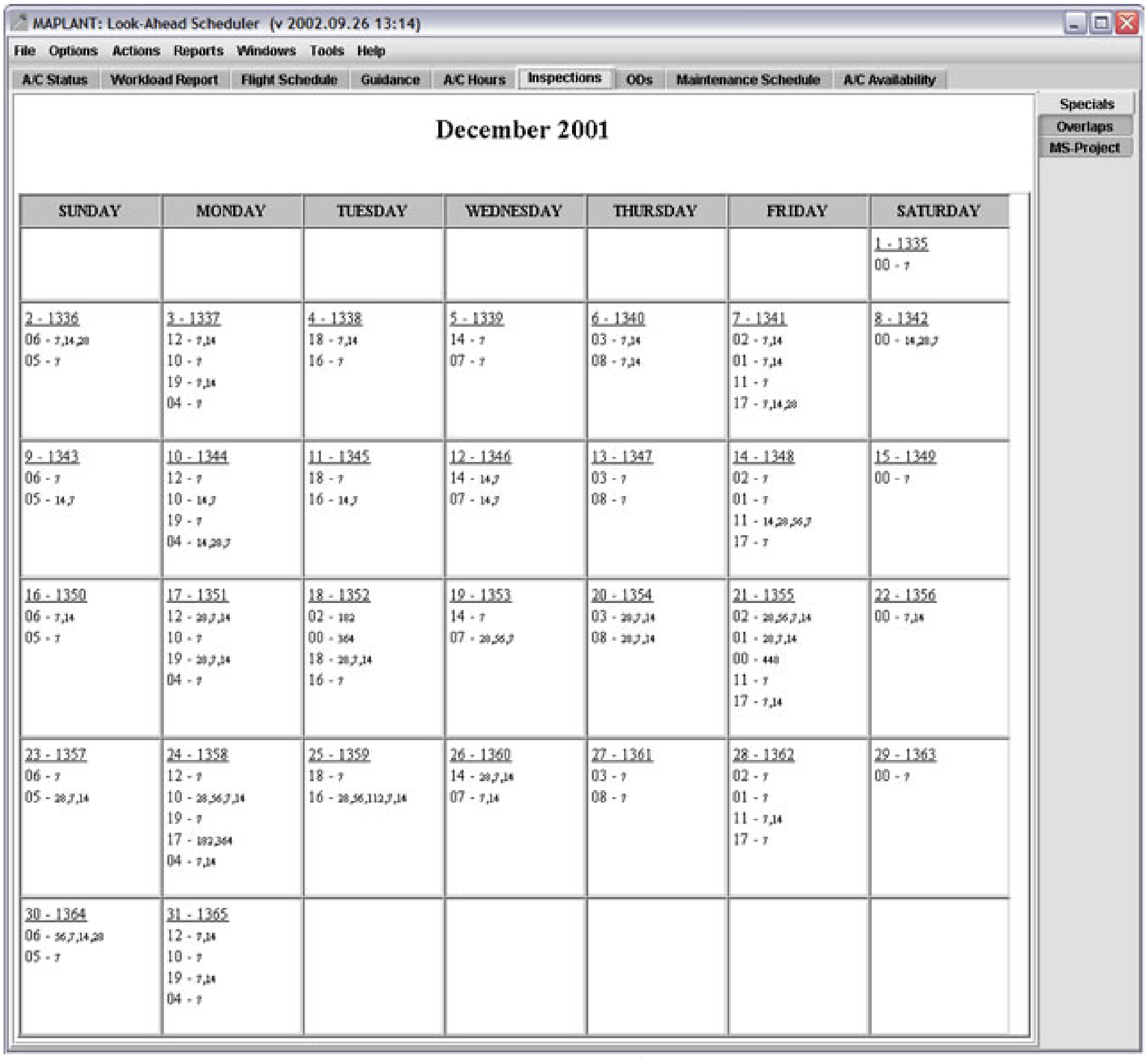


Figure 16. Upcoming Special Inspections

Operational Deviations

Operational deviations are any tasks that aren't tracked by some legacy computer system, such as NALCOMIS. You may use the ODs screen to add work packages to the plan that are additional work not included by repairs and preventive maintenance. For example, if the flight schedule proposed by ops contains missions requiring certain aircraft reconfigurations, such as hanging guns or external fuel tanks, you may track the deadlines for such tasks here.

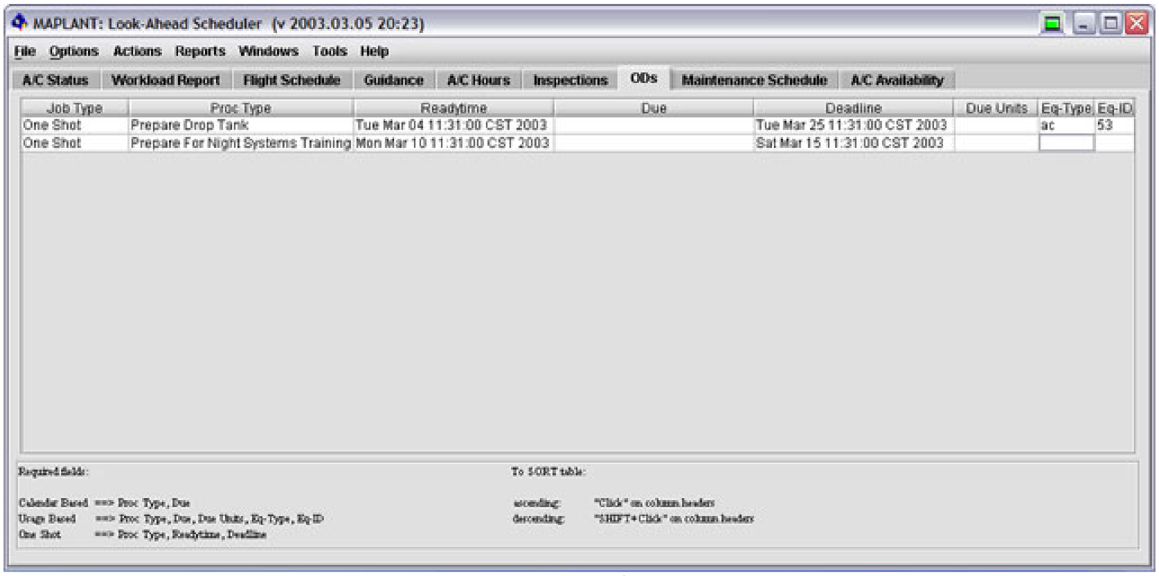


Figure 17. Operational Deviations

Maintenance Schedule

We will see later how to generate a maintenance plan. After going through this process, you may wish to view the details of when the various maintenance actions will be performed. Switching to the *Maintenance Schedule* tab, shown below, will reveal a large button labeled "*Press to View"* in the top half of the window. Pressing this button will invoke Microsoft Project 2000™ and load the generated maintenance schedule. Note that this requires a separate installation of MS-Project. More information about how to manipulate the MS-Project based schedule viewer is presented in later sections of the manual.

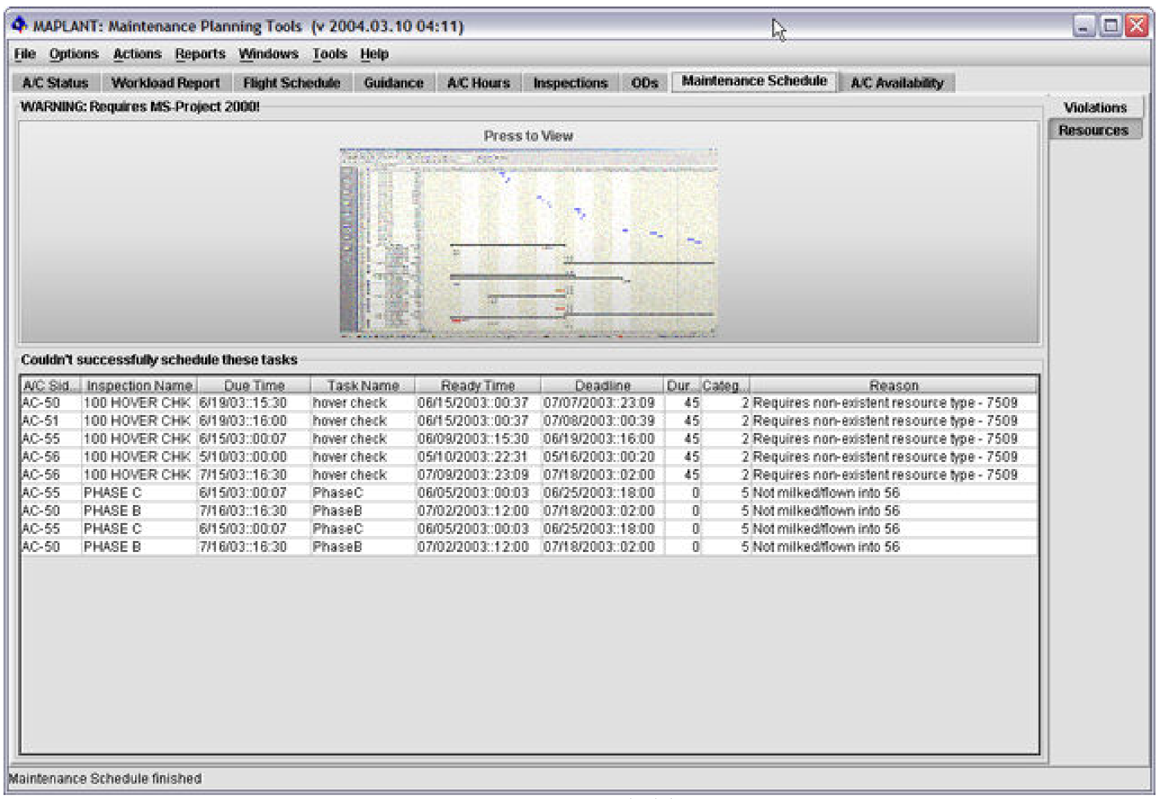


Figure 18. Maintenance Schedule Viewer

The bottom portion of the screen lists any detected discrepancies or issues related to the schedule. In the example above, MAPLANT warns the user that pilots must be reserved for four upcoming test flights and that it was unable to meet the soft constraint of scheduling phase inspections at the same time as nearby 56-dsi inspections.

Resource Usage

Once a maintenance schedule has been successfully computed, this interface can be used to see the number of resources required over time by the plan. The x-axis is time; the y-axis is number of resources. Additionally, the *hotspots* or the riskiest areas of the plan are determined and presented to the user. Pressing any hotspot button will zoom into the associated region of the plot. Be aware that these charts are not recomputed until the *Refresh* button is selected.

In the plots below, the green curves represents available capacity. Notice that it is possible to have different capacity between the day and the night shift (as shown by the square wave curves). The red curve is the numbers of resources actually required by the computed schedule. Finally, the blue curve may be used to estimate resource usage requirements before an actual schedule has been determined. This is helpful for predicting bottlenecks caused by lack of required resources.

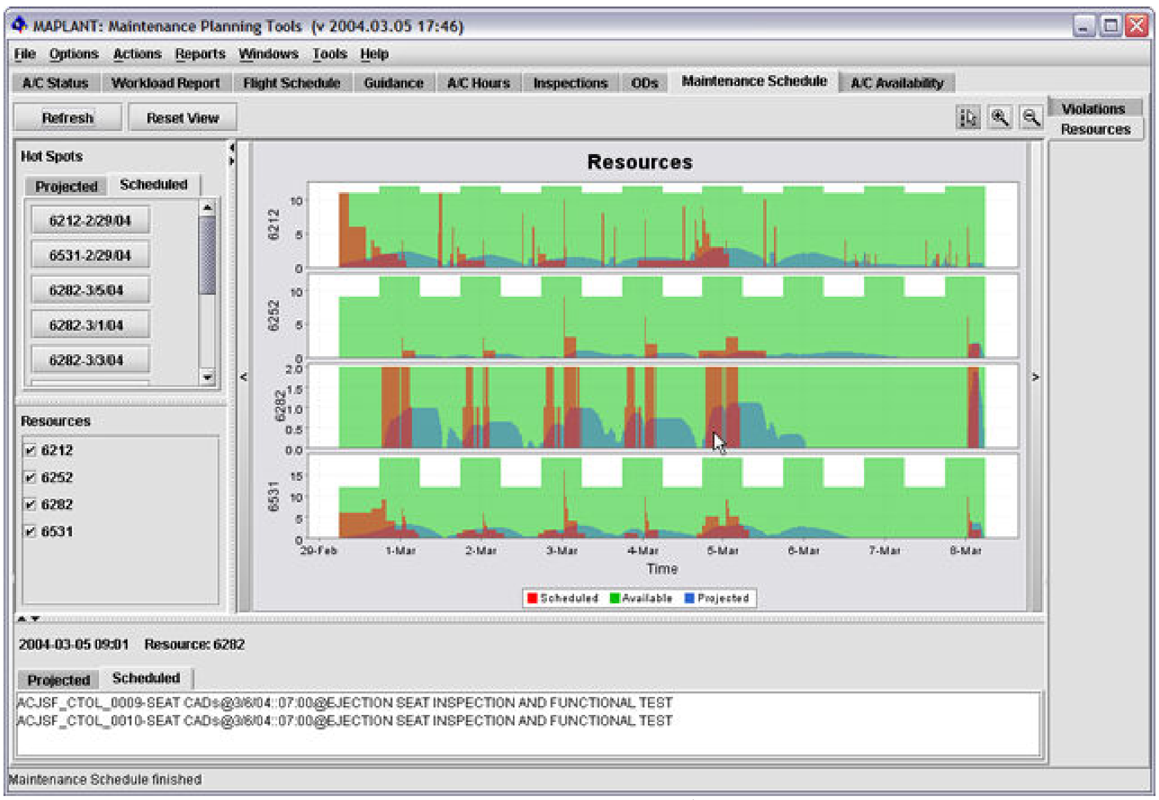


Figure 19. Resource Usage Analysis

As you move your mouse over the various areas of the chart, the currently selected point is shown near the bottom left section of the screen. In the example above, the cursor is pointing to *2004-03-05 09:01* for resources of type *6282*. Additionally, all of the individual tasks contributing to this coordinate are listed in the lower split pane. This task list is updated dynamically as you move the cursor. The toolbar in the top right area of the screen is used to switch between different functions that are invoked when left-clicking on a chart (e.g. zooming). Finally, right-clicking on any chart results in a pop-up menu where several other functions are available.

Aircraft Availability

Similar to the *Maintenance Schedule* tab, *A/C Availability* contains a button that invokes an external viewer, which displays (a) the projected number of healthy aircraft throughout the planning period and (b) the aircraft requirements of the proposed flight schedule.

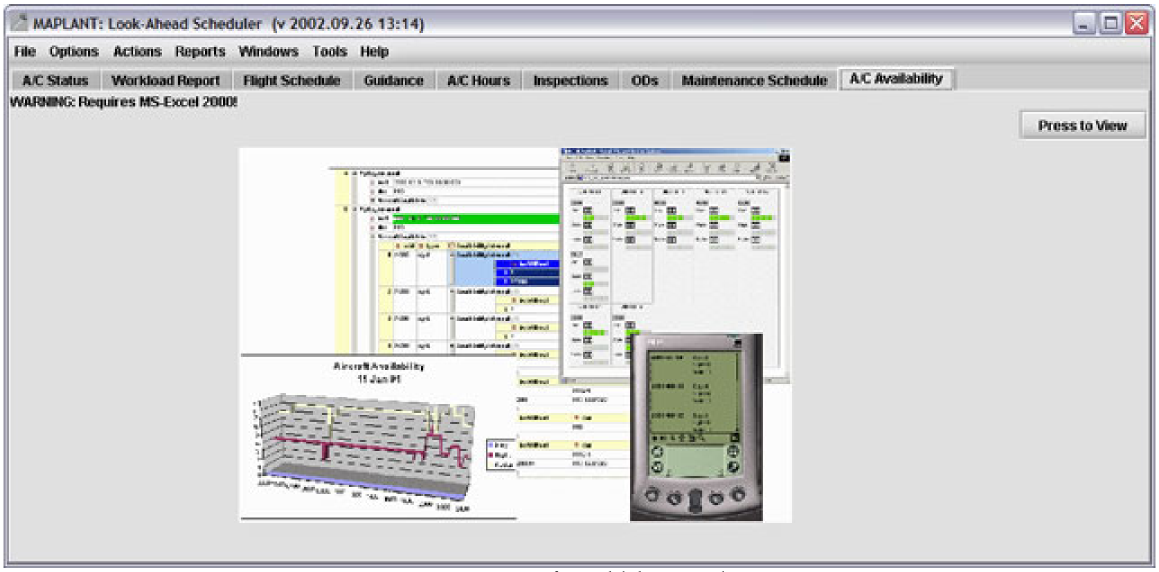


Figure 20. Aircraft Availability Launcher

In this case, the third-party program is Microsoft Excel 2000™. This particular view, however is but one of many forms that the aircraft availability information may take. For example, there is a non-graphical version that is consumed by SNAP. Or, one might wish to export a version of the report that could be displayed on a simple PDA device or in a standard web browser.

The Menu System

Basic Menu Items

There are seven menu categories in the menu bar at the top of the MAPLANT application. From left to right, they are *Files*, *Options*, *Actions*, *Reports*, *Windows*, *Tools*, and *Help*.

Figure 21. The Seven Menu Items in MAPLANT

The *File* Menu

This menu contains menu items for basic file manipulation and system shutdown. Certain items may be disabled depending upon which aspect of the system is the current focus.

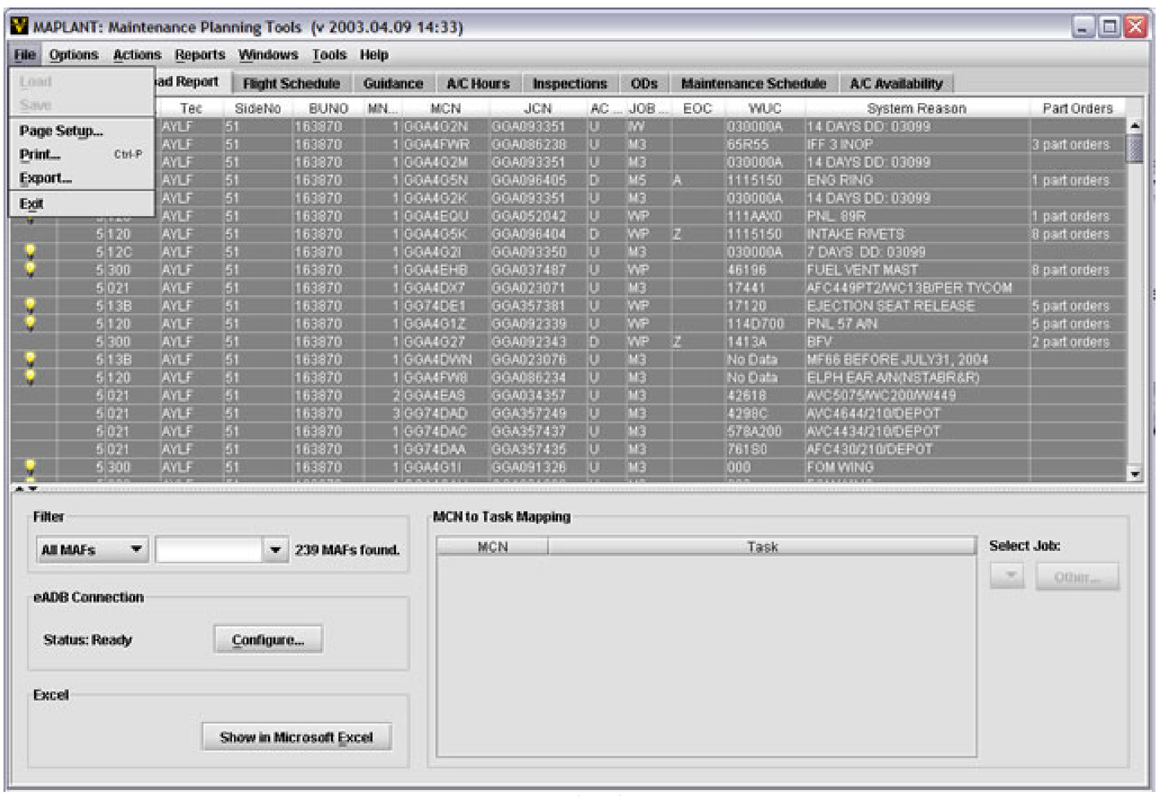


Figure 22. The File Menu

Load

The *Load* command is only enabled when either *Guidance* or the *Flight Schedule* views are active. These actions should only be attempted after completing the *Saving and Loading Guidance* and the *Round-Trip Planning with SNAP* tutorials in the *Advanced Users* section of this manual.

Save

The next menu item is the *Save* command. This allows one to save a raw copy of the currently selected dataset. The currently selected dataset is determined by the current view tab. This item is enabled for *aircraft status*, *flight schedule*, *guidance*, *inspections* and *ODs*. Additionally, the save command will be enabled for the *maintenance schedule* and for the *aircraft availability* tabs if a maintenance schedule has already been computed.

Page Setup

This menu item allows one to select an available print device and to set the paper handling characteristics for print jobs (e.g. landscape versus portrait paper orientation).

Print

Prints the currently selected view. Printing of large datasets, such as the *workload report* or the *maintenance schedule*, should use the provided *export to MS-Excel/MS-Project* functionality and print from within that application instead.

Export

The *Export* command will save the current state of your MAPLANT session to a file. This file will sometimes be required by technical support in order to reproduce your exact situation. If requested to do so, please export your MAPLANT session to a temporary file on your local machine and send as an attachment to maplant-support@isis-server.isis.vanderbilt.edu. See also the *Getting Help* chapter.

Exit

Shuts down the MAPLANT application.

The *Options* Menu

The *Options* menu contains two items. The *Timezone* feature allows one to switch all displayed dates between the format specified by the local computer's time zone setting and that of Greenwich Mean Time. The *Log Level* menu allows one to change the verbosity level of the messages that MAPLANT writes to its log files. These log files are kept in a subdirectory of the %MAPLANTDIR% installation directory.

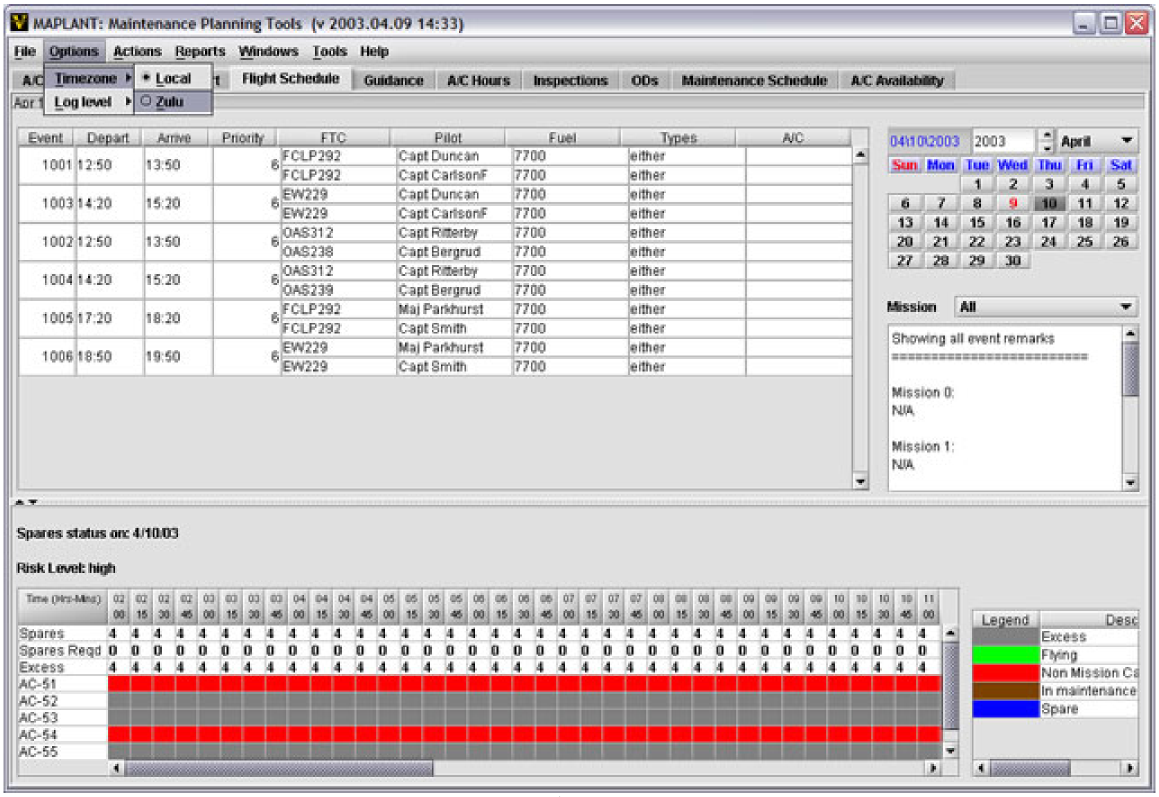


Figure 23. The Options Menu

The *Windows* Menu

MAPLANT may have more than one window displayed at any time. You may open a new window with the *New Window* menu item. Window may be close by left-clicking with a mouse in the "X" icon in the top right corner of the window or with the *Close Current* menu item. If there is only one window, closing it is equivalent to the *File/Exit* action. Also, the previously dismissed *MAPLANT Log Window* may be brought back to the desktop by selecting the *Show Log* menu item.

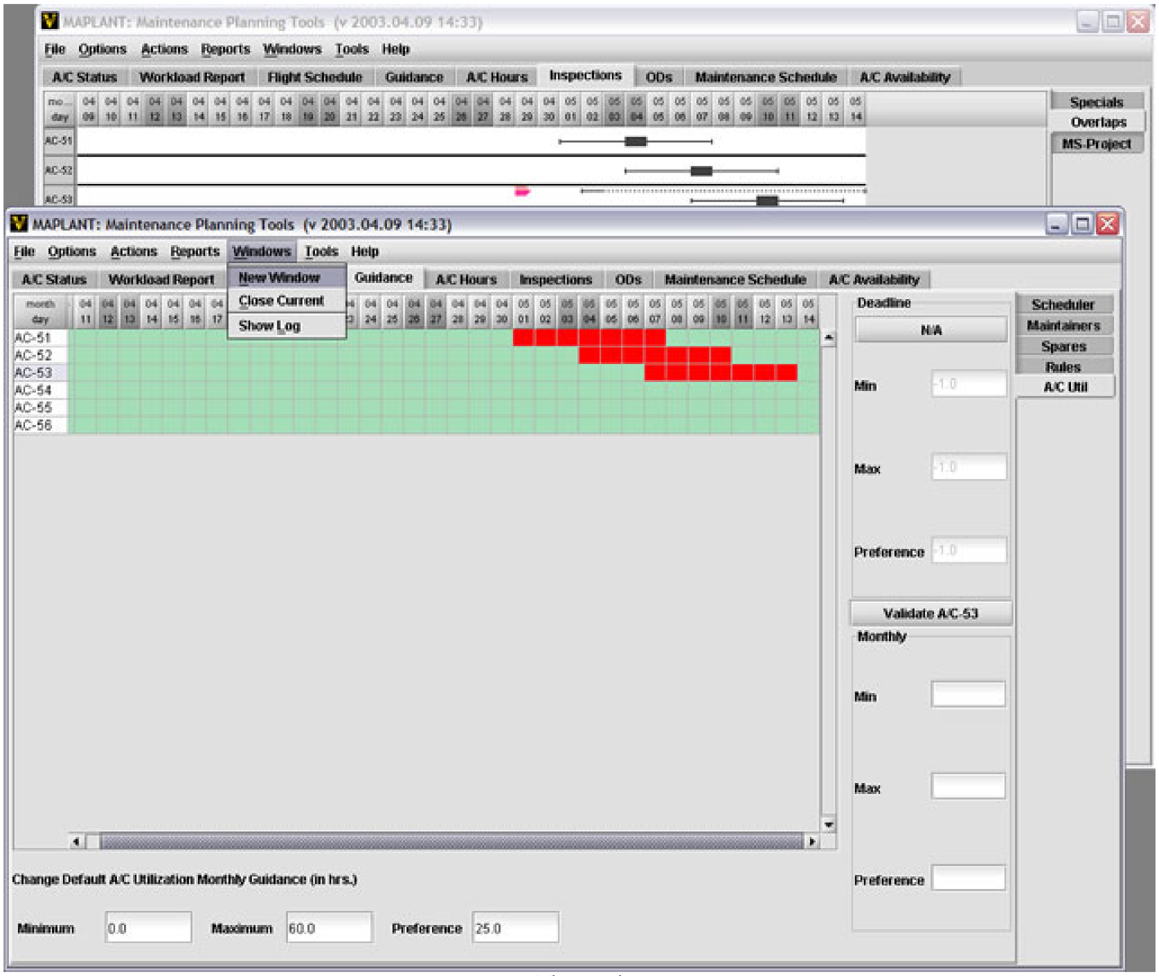


Figure 24. The Windows Menu

The *Help* Menu

Contents will display this user's manual, and *About MAPLANT* will display version and technical support options details.

Figure 25. The Help Menu

The Actions Menu

There are two menu items listed under the *Actions* menu. Each will initiate its corresponding scheduling algorithms. The *Flight Schedule Assignment* scheduler will assign available aircraft to the sorties on the flight schedule in a way that obeys all laws and guidance specified by the user. The details of the resulting flight schedule assignments may be view with the *Flight Schedule* tab and the *A/C Hours* tab.

The *Compute Maintenance Schedule* action will arrange all maintenance actions in time such that (a) laws are unbroken, (b) guidance or user preferences are taken into account and (c) available resources are not overconsumed. Once a maintenance schedule has been computed, its can be examined under the *Maintenance Schedule* tab or the bottom portion of the split screen under the *Flight Schedule* view. Additionally, once both schedules have been computed, aircraft availability is forecasted and can be view in the *A/C Availability* tab.

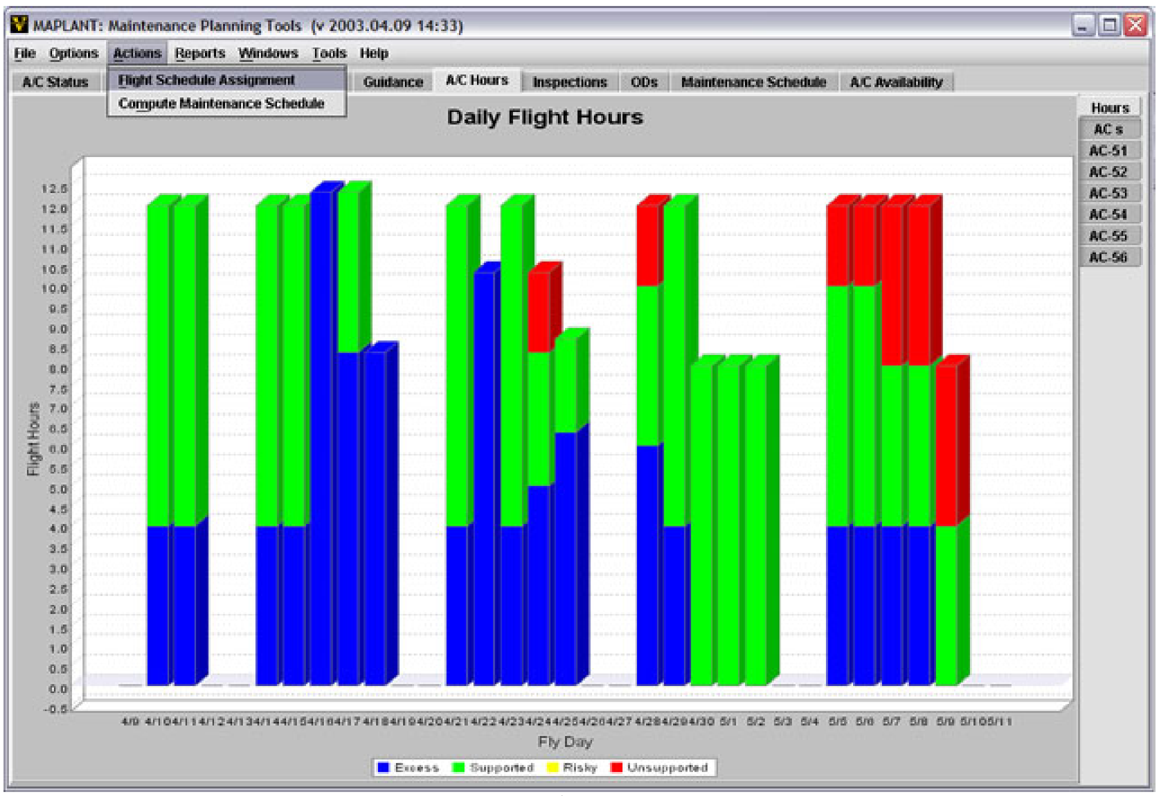


Figure 26. The Actions Menu

The Reports Menu

This menu allows access to several useful summary reports. The *MMP-Inspections* command will quickly produce a list of the scheduled calendar-based inspections for the upcoming month. The contents of this file (see title bar in the report view for filename) may be cut and pasted into Microsoft Word. Similarly, the *MMP-Roster* will generate a list of all personnel and their qualifications grouped by work center.

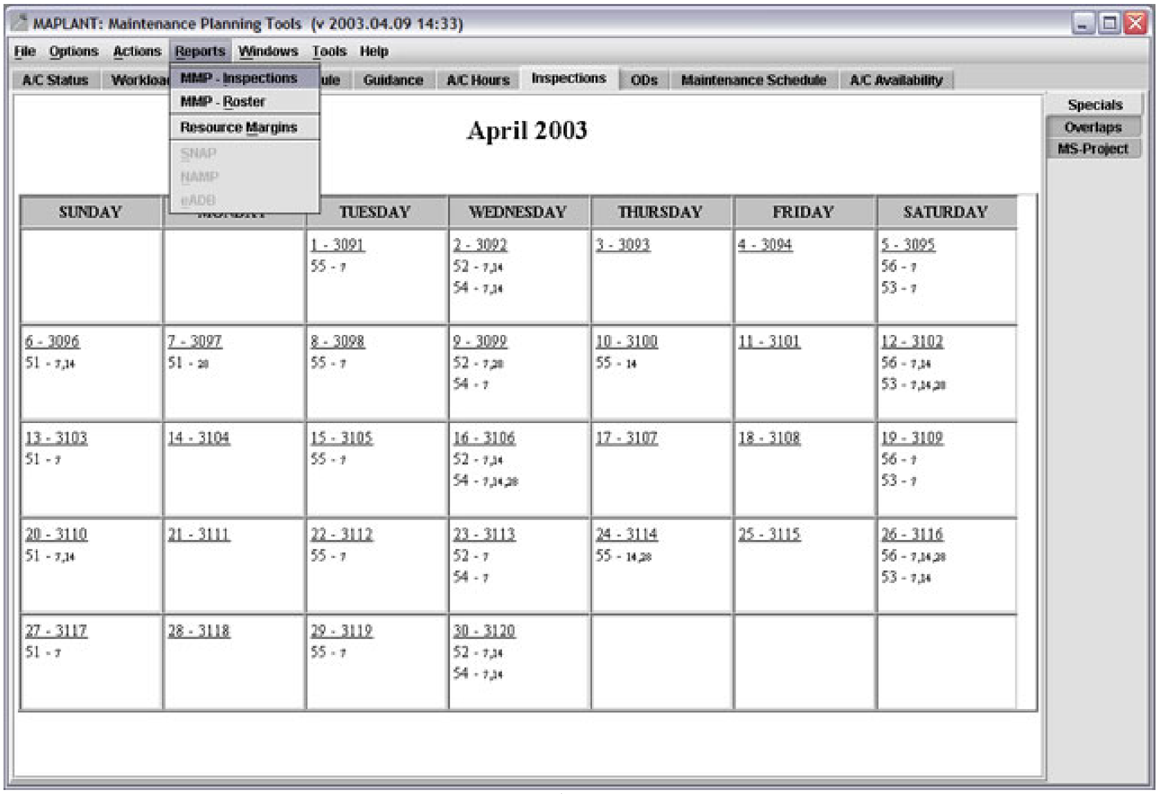


Figure 27. The Reports Menu

The *Resource Margins* report will display the top-ten list of most consumed *resource-category-days*. You must, of course, have already completed the process of computing an actual maintenance schedule. The following figure shows an example of this report. The yellow portion of the graph shows capacity and the green curve denotes resources consumed by the computed maintenance schedule. This report says that the usage of resource type 6282 on day 12-Mar-2003 is the eighth most dangerous aspect of the maintenance schedule. Note that this schedule was computed with a scheduling engine that allows overtime within reason. This is why the *used* curve (green) spikes above the *available* curve (yellow) for a short time frame. This is an indicator that the shift definitions might need to be tweaked for that particular work day. Categories lacking a yellow curve indicate resources that aren't being actively managed in this particular schedule. In the case below, management of support equipment had been disabled. The green *resources required* graphs are plotted in either case.

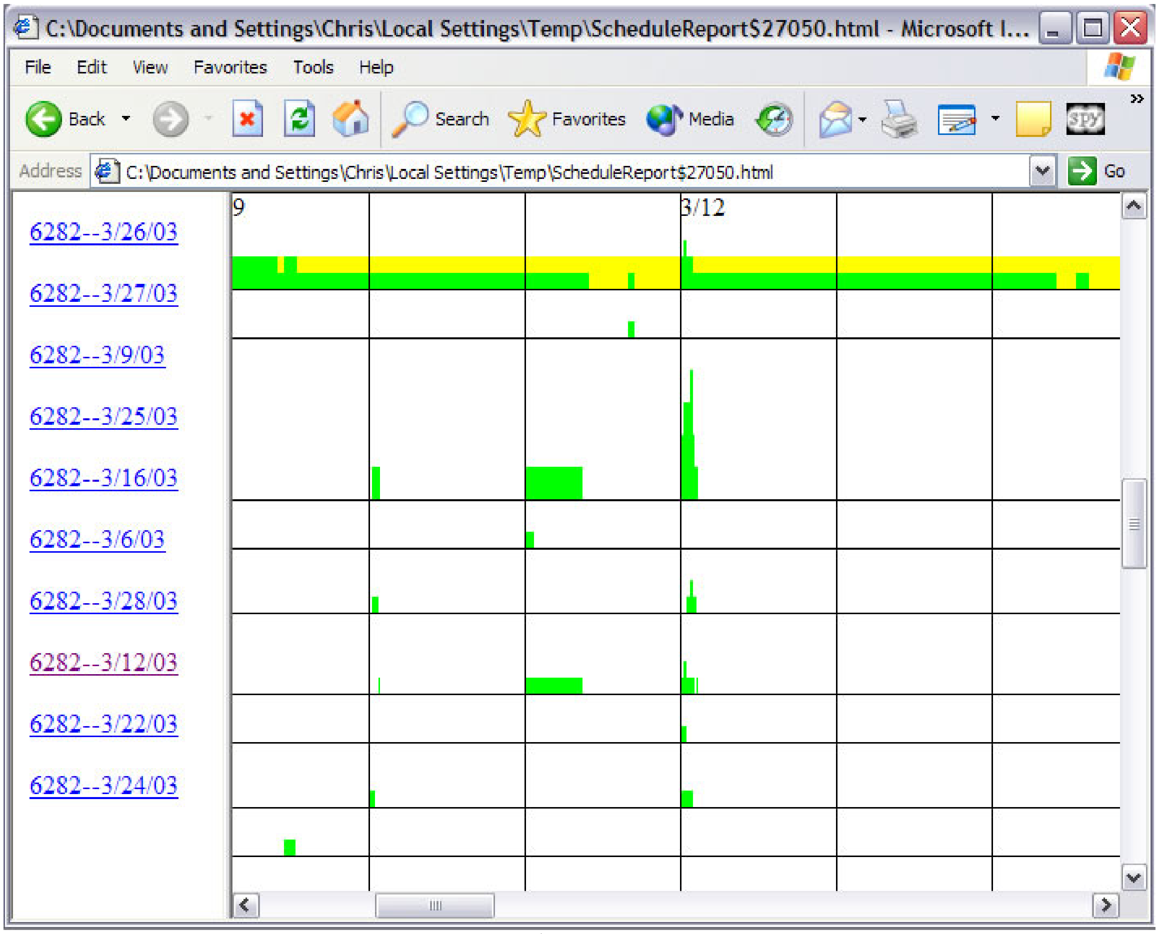


Figure 28. Example Resource Margins Report

The *SNAP* report menu item is used to generate an aircraft availability feedback for the SNAP flight scheduling tool. This will identify flydays where excess capability exists as well as proposed missions which must be cancelled or modified due to unsupportability. This command is only enabled after a valid maintenance schedule has been computed.

Figure 29. SNAP Report Generation

The *NAMP** reports menu will connect you to the reporting facility of the *Data Warehouse*.

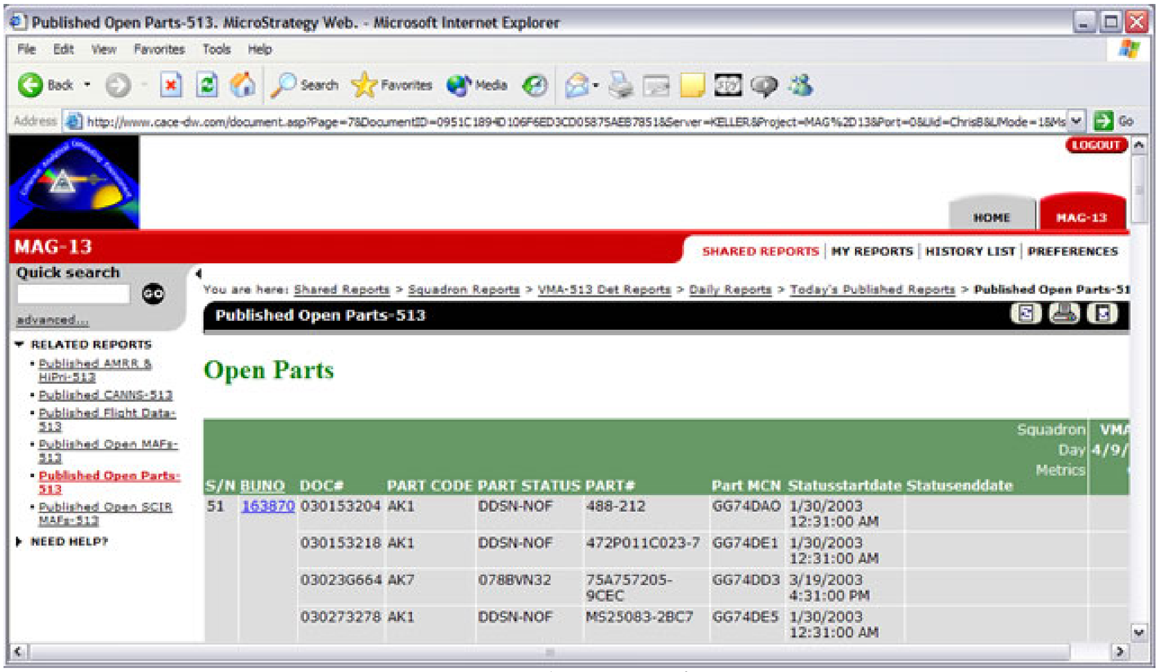


Figure 30. The NAMP Interface

Similarly, the *eADB** menu item will instantiate a session with the *Electronic Aircraft Discrepancy Book* tool.

Figure 31. The eADB Interface

* Both NAMP reporting and the eADB are disabled for demonstration versions of the software.

The *Tools* Menu

The menu items under tools provide an alternative to using the programs icons described previously under the "Other MAPLANT Tools and Icons" section of this manual. See the "Starting MAPLANT" chapter under "Getting Started".

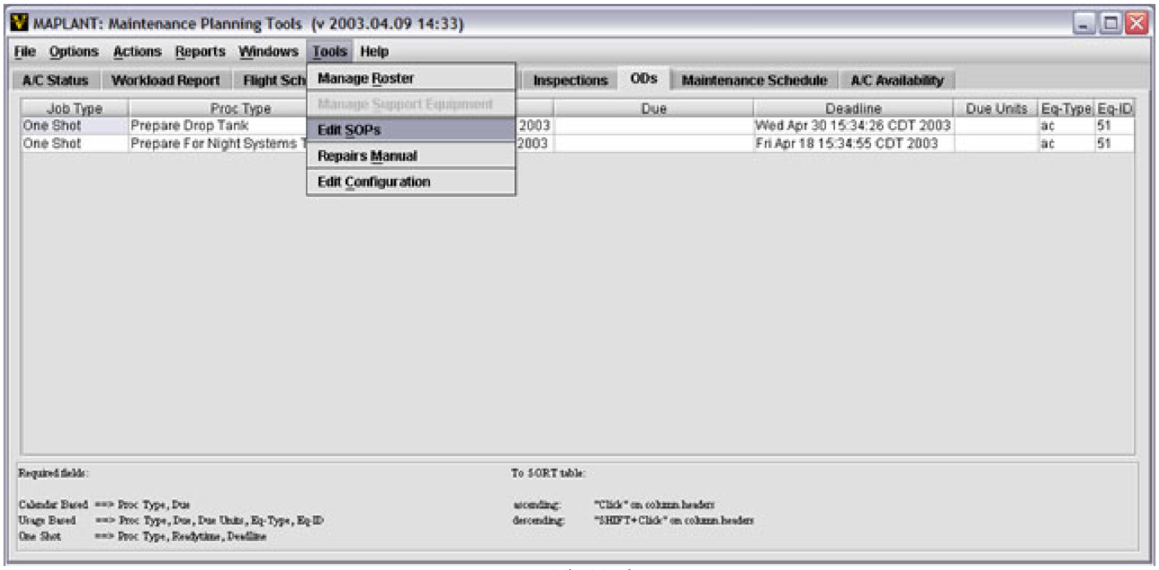


Figure 32. The Tools Menu

On-Line Help

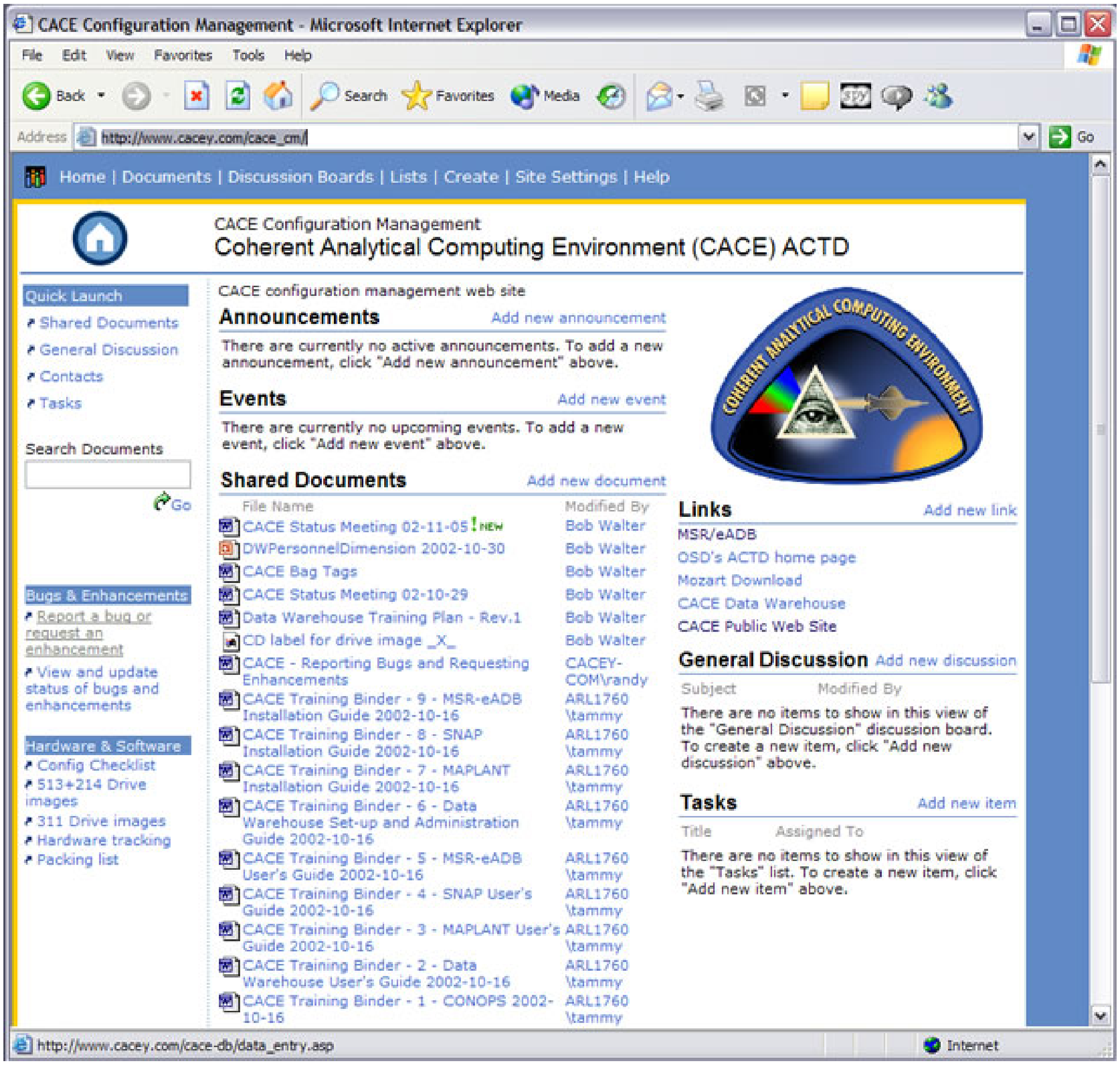
This manual is accessible from the MAPLANT *Help* menu or from the help icon in your programs folder.

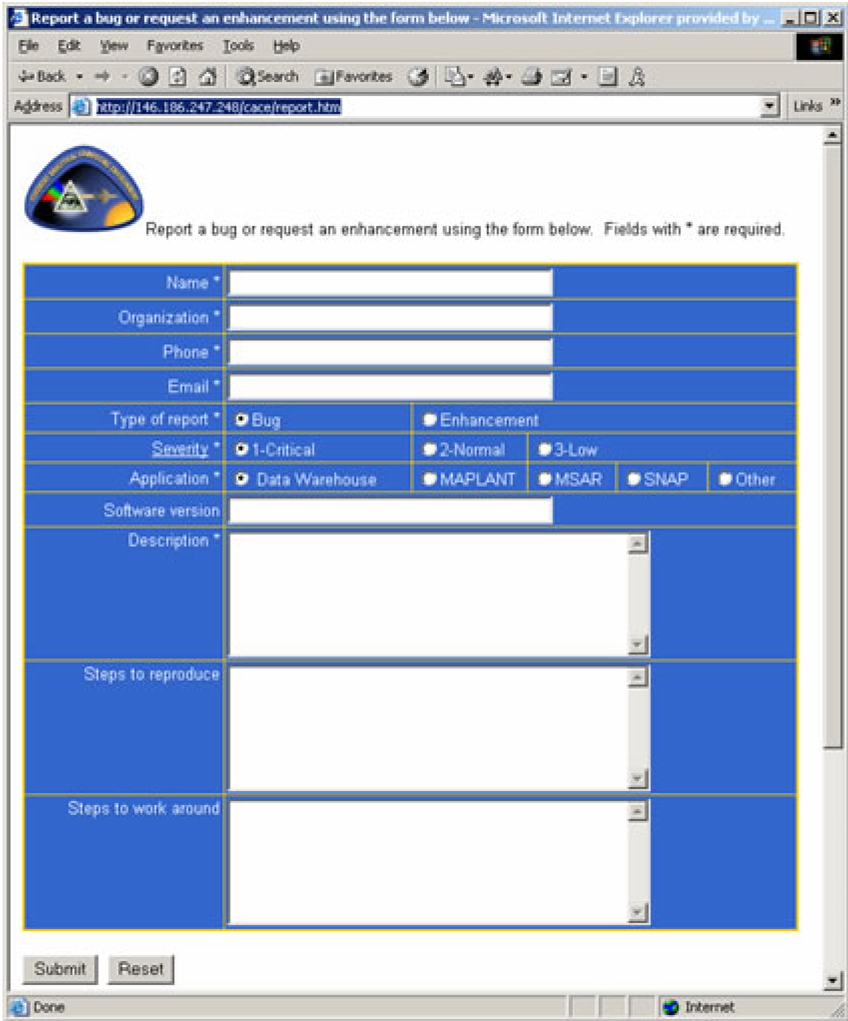
Requesting Technical Support

If you encounter problems that require access to the developers of MAPLANT, please send email containing a detailed description of the problem and what you were attempting to do at the time to maplant-support@isis.vanderbilt.edu. If possible, also attach the .MZIP file that can be created by using the File/Export menu item.

CACE - How to Report Bugs and Request Enhancement Requests

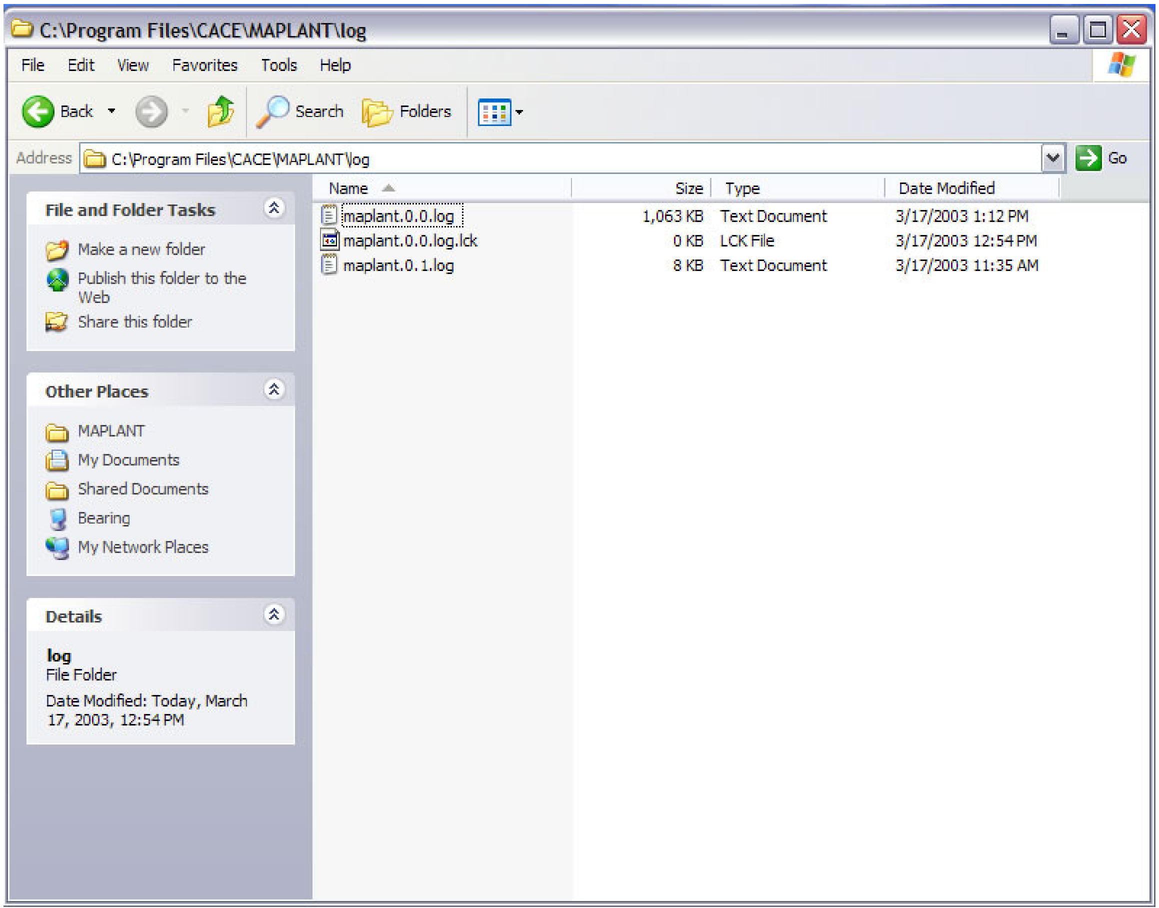
1. Connect to the CACE Configuration Management web site at: http://www.cacey.com/cace_cm/
2. Enter your username and password:
   username mag13
   password flyguys
3. Click the"*Report a bug or request an enhancement*" link in the left column and complete and submit the form.


CACE Configuration Management - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://www.cacey.com/cace_cm/
Go
Home | Documents | Discussion Boards | Lists | Create | Site Settings | Help
CACE Configuration Management
Coherent Analytical Computing Environment (CACE) ACTD
Quick Launch
Shared Documents
General Discussion
Contacts
Tasks
Search Documents
Go
Bugs & Enhancements
Report a bug or request an enhancement
View and update status of bugs and enhancements
Hardware & Software
Config Checklist
513+214 Drive images
311 Drive images
Hardware tracking
Packing list
CACE configuration management web site
Announcements
Add new announcement
There are currently no active announcements. To add a new announcement, click "Add new announcement" above.
Events
Add new event
There are currently no upcoming events. To add a new event, click "Add new event" above.
Shared Documents
Add new document
File Name
Modified By
CACE Status Meeting 02-11-05 ! NEW
Bob Walter
DWPersonnelDimension 2002-10-30
Bob Walter
CACE Bag Tags
Bob Walter
CACE Status Meeting 02-10-29
Bob Walter
Data Warehouse Training Plan - Rev.1
Bob Walter
CD label for drive image _X_
Bob Walter
CACE - Reporting Bugs and Requesting Enhancements
CACEY-COM\randy
CACE Training Binder - 9 - MSR-eADB Installation Guide 2002-10-16
ARL1760\tammy
CACE Training Binder - 8 - SNAP Installation Guide 2002-10-16
ARL1760\tammy
CACE Training Binder - 7 - MAPLANT Installation Guide 2002-10-16
ARL1760\tammy
CACE Training Binder - 6 - Data Warehouse Set-up and Administration Guide 2002-10-16
ARL1760\tammy
CACE Training Binder - 5 - MSR-eADB User's Guide 2002-10-16
ARL1760\tammy
CACE Training Binder - 4 - SNAP User's Guide 2002-10-16
ARL1760\tammy
CACE Training Binder - 3 - MAPLANT User's Guide 2002-10-16
ARL1760\tammy
CACE Training Binder - 2 - Data Warehouse User's Guide 2002-10-16
ARL1760\tammy
CACE Training Binder - 1 - CONOPS 2002-10-16
ARL1760\tammy
COHERENT ANALYTICAL COMPUTING ENVIRONMENT
Links
Add new link
MSR/eADB
OSD's ACTD home page
Mozart Download
CACE Data Warehouse
CACE Public Web Site
General Discussion
Add new discussion
Subject
Modified By
There are no items to show in this view of the "General Discussion" discussion board. To create a new item, click "Add new discussion" above.
Tasks
Add new item
Title
Assigned To
There are no items to show in this view of the "Tasks" list. To create a new item, click "Add new item" above.
http://www.cacey.com/cace-db/data_entry.asp
Internet


Report a bug or request an enhancement using the form below - Microsoft Internet Explorer provided by ...
File Edit View Favorites Tools Help
Back Search Favorites
Address http://146.186.247.248/cace/report.htm
Links
Report a bug or request an enhancement using the form below. Fields with * are required.
Name *
Organization *
Phone *
Email *
Type of report * Bug Enhancement
Severity * 1-Critical 2-Normal 3-Low
Application * Data Warehouse MAPLANT MSAR SNAP Other
Software version
Description *
Steps to reproduce
Steps to work around
Submit Reset
Done
Internet

The MAPLANT Log file

MAPLANT constantly writes logging information to a text file in the CACE directories as it runs. These files are sometimes needed when debugging a problem with MAPLANT. If asked to do so, please navigate to the appropriate directory (e.g. C:\Program Files\CACE\MAPLANT\logs) and send all files as email attachments to maplant-support@isis.vanderbilt.edu.



Licensing MAPLANT

By default, MAPLANT comes configured with some sample, pre-defined, demonstration datasets. In order to enable the tool for actual deployment, you will need to use the *license* icon (shown above) to enter your installation's site-specific license key. Please contact maplant-support@isis.vanderbilt.edu if you need help obtaining this key.

The Configuration Editor

The "Edit Configuration" icon is available from your MAPLANT programs icons or via the *Tools* menu when the main MAPLANT application is running. This is used to modify configuration settings and system preferences for the software. The interface is divided into two main sections. Along the left is a tree browser used to navigate to various categories of configuration settings. The right pane shows the collection of settings for the chosen category.

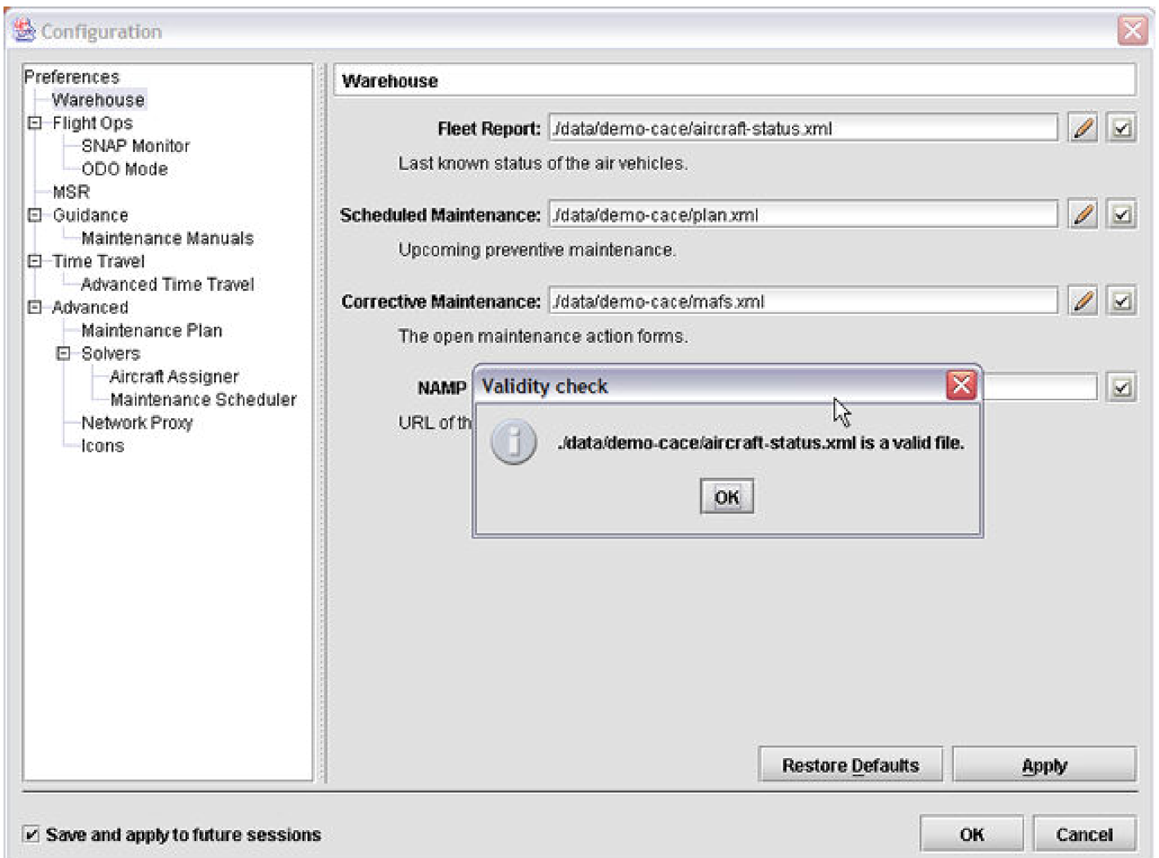


Figure 33. The Configuration Editor

Individual settings will have a short name and an associated longer description. Additionally, each setting will have associated buttons that are customized to the data type of the property. In the screenshot above, all entries may either contain a file name or a URL. In this case, there are two associated buttons to aid in the setting of the property's value. The pencil icon may be used to browse the local file system for a valid file. The button with the checkmark icon will verify that the chosen path is either a readable file or a valid URL.

The checkbox in the lower left area of the screen is used to select between temporary modifications that should only be in effect for the current MAPLANT session, and longer term changes that will change the default behavior for all future planning sessions.

Configuring the Proxy Server

The ability to run MAPLANT through a Microsoft Proxy Server is temporarily unsupported. If you have a need to re-enable this facility, please contact maplant-support@isis.vanderbilt.edu.

Flight Schedules and SNAP

MAPLANT receives all flight schedule information directly from the operations advisor tool called SNAP. There are two techniques for importing new or updated information about upcoming flights, which are discussed next.

Importing New Flight Schedules

In a production environment, your CACE administrator will have configured both SNAP and MAPLANT to be connected to each other at run-time. When SNAP publishes new flight schedules or modifies the details of existing schedules, MAPLANT will automatically recognize this and inform the user that new schedules are available.

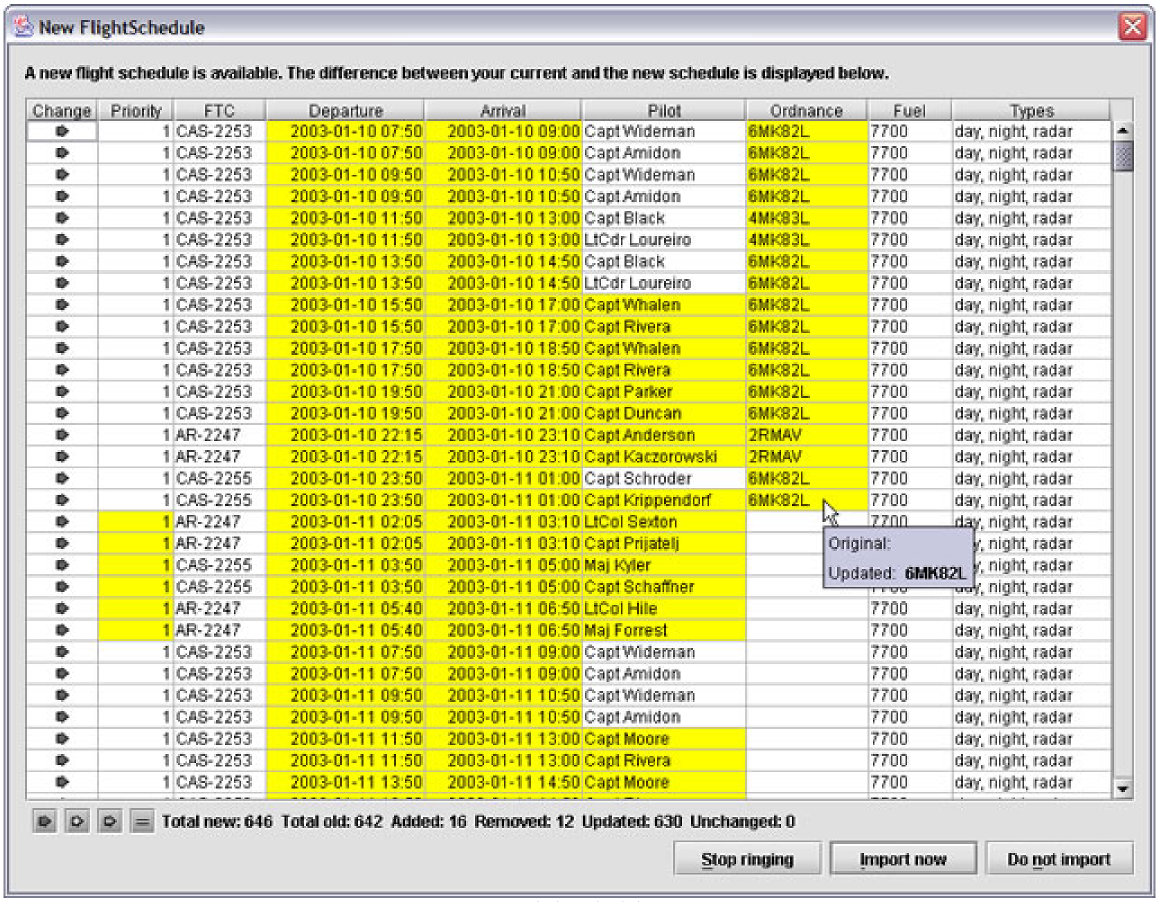


Figure 34. Flight Schedule Merging

The differences between the old and new schedules will be highlighted and the user is asked whether the new schedules should be merged into the current planning session.

Selecting and Merging Schedules

Alternatively, you my manually merge existing flight schedules using an external utility found under the *Tools* section of your icons.

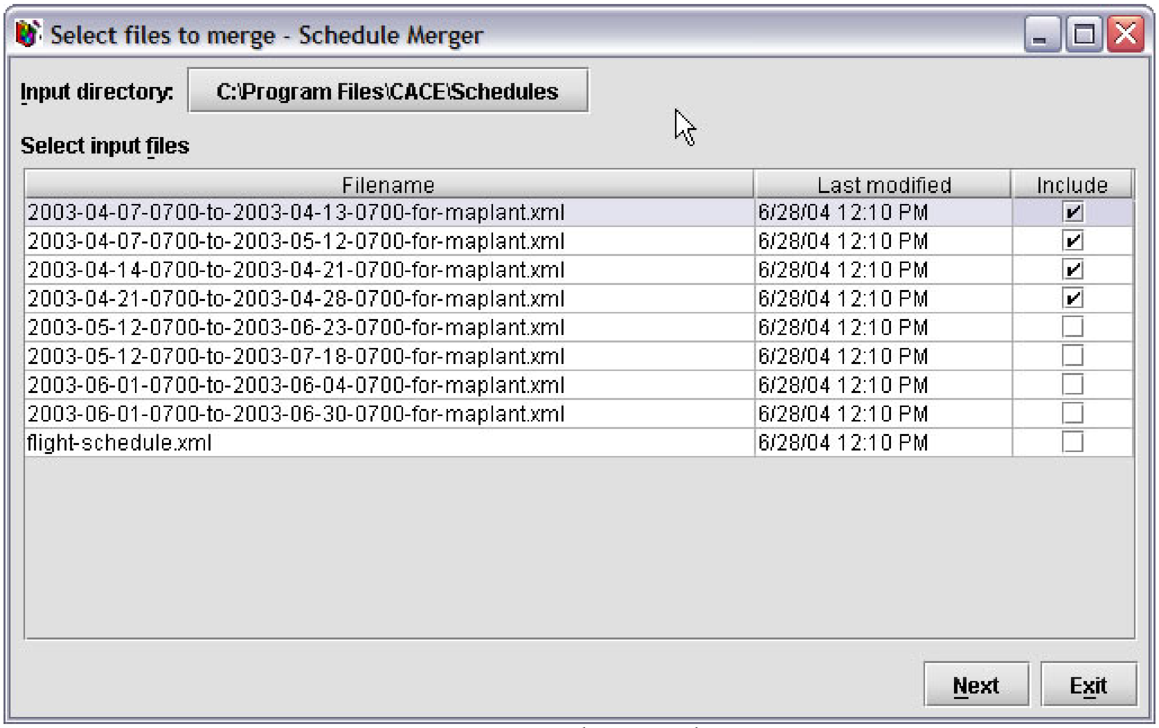


Figure 35. Manual Merging Utility

In the example screen shot above, all flight schedules relating to the time period from Apr-07th to May-12th will be merged, resulting in a four week plan.

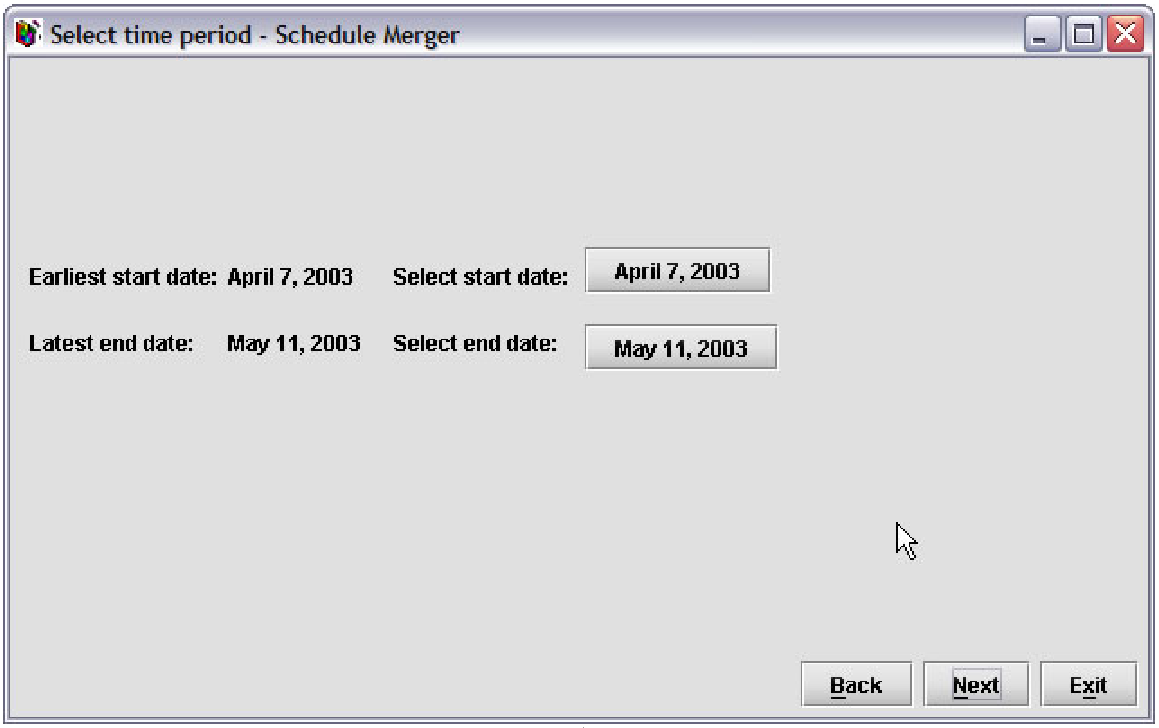


Figure 36. Manual Merging: Page 2

By default, the first and the last flyday will be chosen as the boundaries of the planning horizon. You may override, if desired.

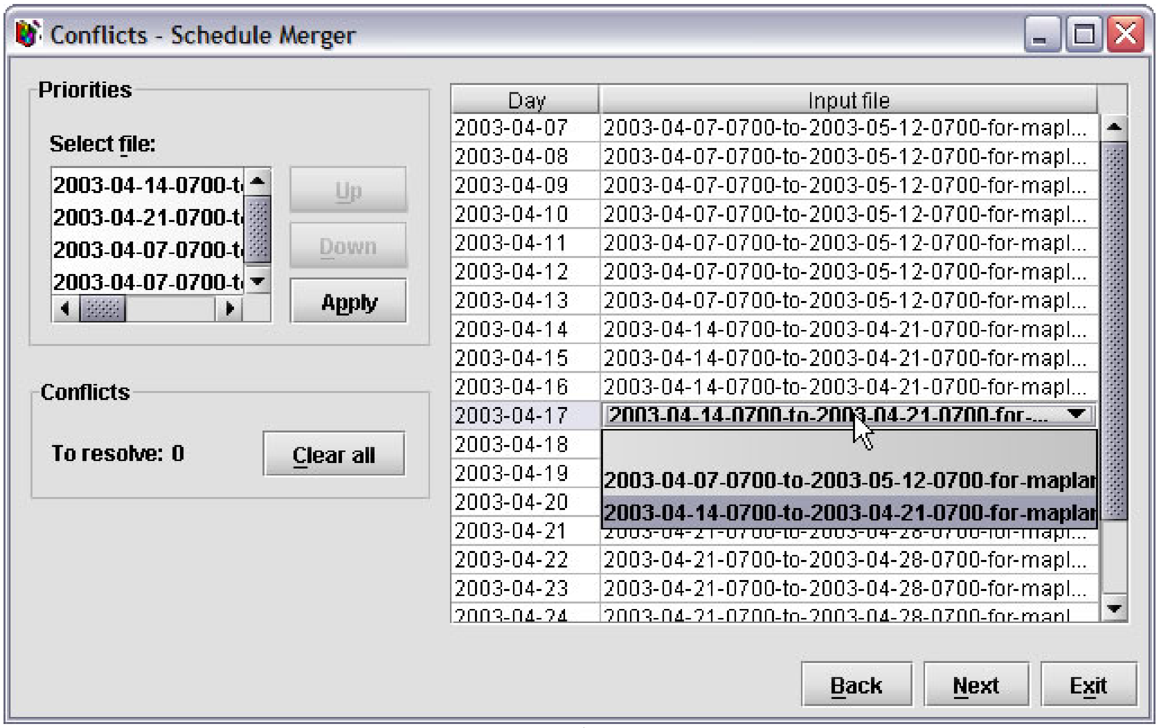


Figure 37. Manual Merging: Page 3

It is possible for multiple input files to contain information for a particular flyday. For example, if you are merging a monthly plan, a weekly plan, and a daily plan, there will be three references to the next flyday and two references to the following six days. Pressing the *Apply* button will result in the default behavior for choosing which version contains an authoritative version for these conflicting flyday records. By default, newer schedules always override older or previous versions of the plan. However, you can manually change this default behavior for individual flydays by clicking in a cell of the *Input File* column on the table to the right.

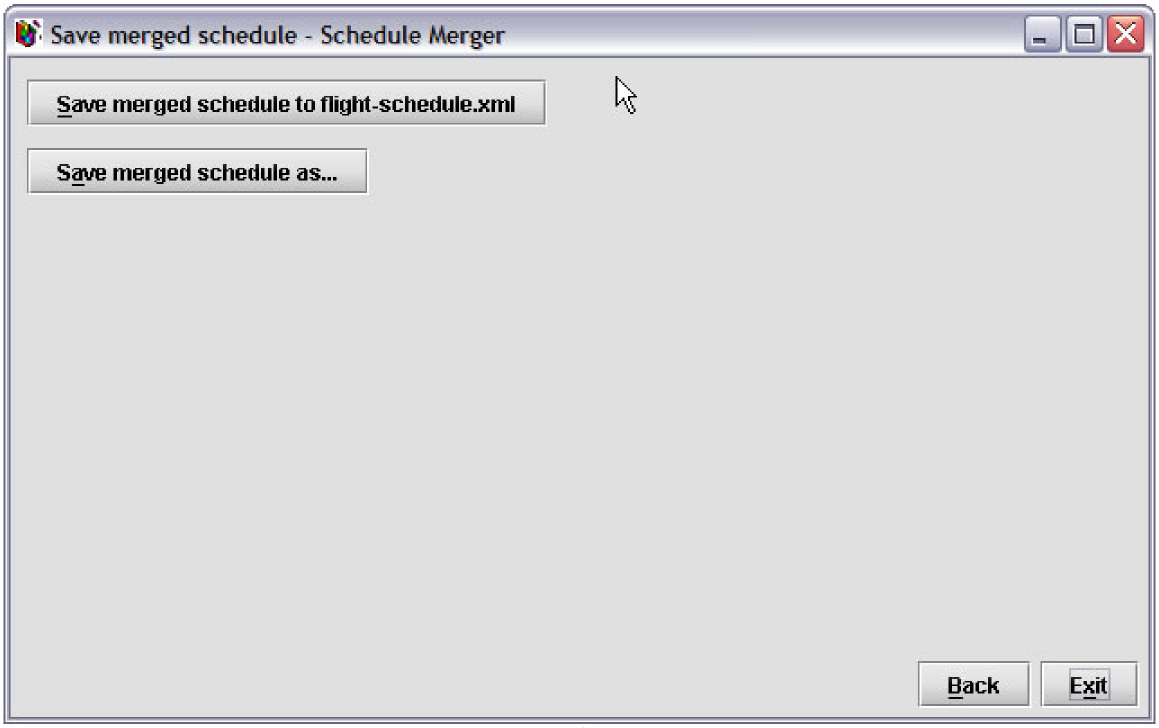


Figure 38. Manual Merging: Page 4

The top button on the last page will save the new set of merged schedules to the default location that MAPLANT will look for the next time it is initialized. The second button allows you to specify the name and location of the output file (for advanced users).

Exporting Aircraft Availability

Availability and SNAP

Just as MAPLANT needs flight schedules from SNAP in order to plan precisely, SNAP requires information about available aircraft from MAPLANT in order to ensure that the operational schedules are realistic.

After completing a MAPLANT planning session, the system can now make projections about the availability of individual air vehicles based on precise knowledge of upcoming maintenance requirements. These projections are published using the *Reports* menu as described under the menu section of *Basic User Interface Components* in the *Getting Started* chapter.

Figure 39. SNAP Report Generation

Repairing Aircraft

While MAPLANT will refuse to automatically assign downed aircraft to the schedule, the user may force an NMC aircraft into the schedule.

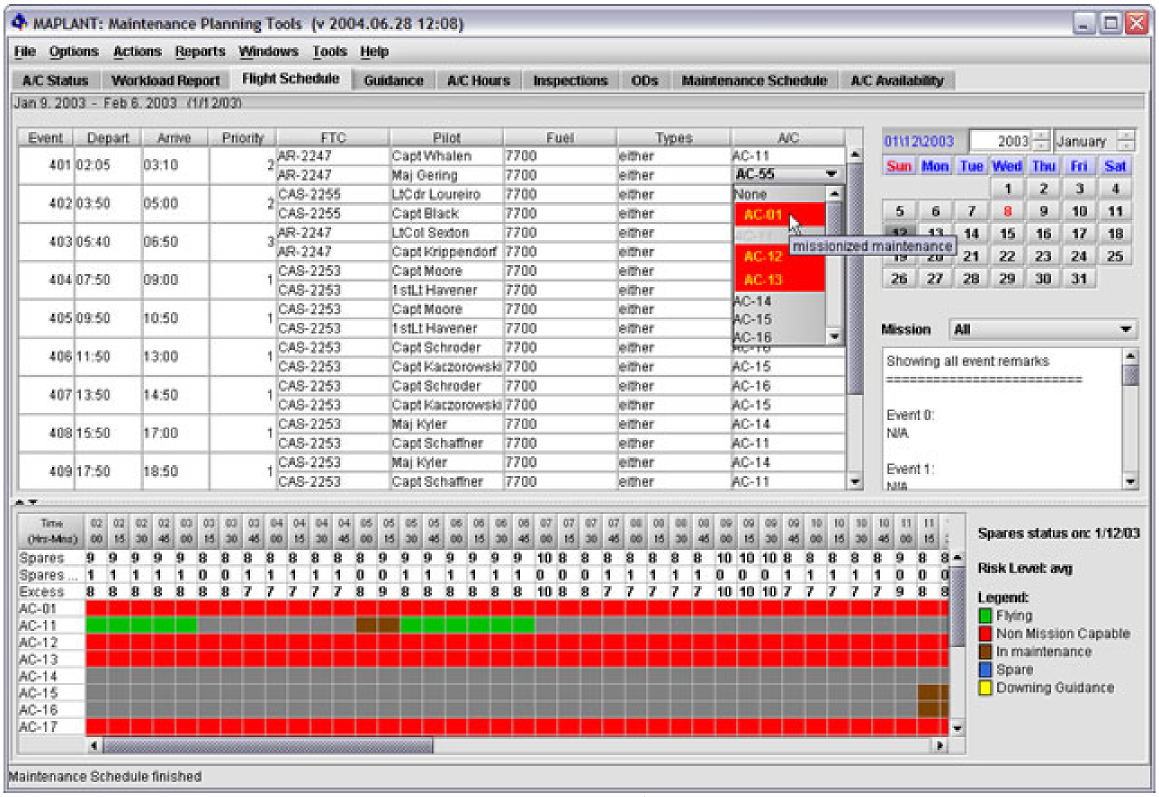


Figure 40. Assigning Hard Downers

Above, in the drop down list of aircraft eligible to fly the chosen mission, AC-01 is flagged as unsuitable due to the current status or health of one or more subsystems. Manually specifying this aircraft assignment will trigger MAPLANT to consult the *Aircraft Suitability Agent*, one of the key services provided by the MSR component (see *CACE Software* under the *Concept of Operations* chapter).

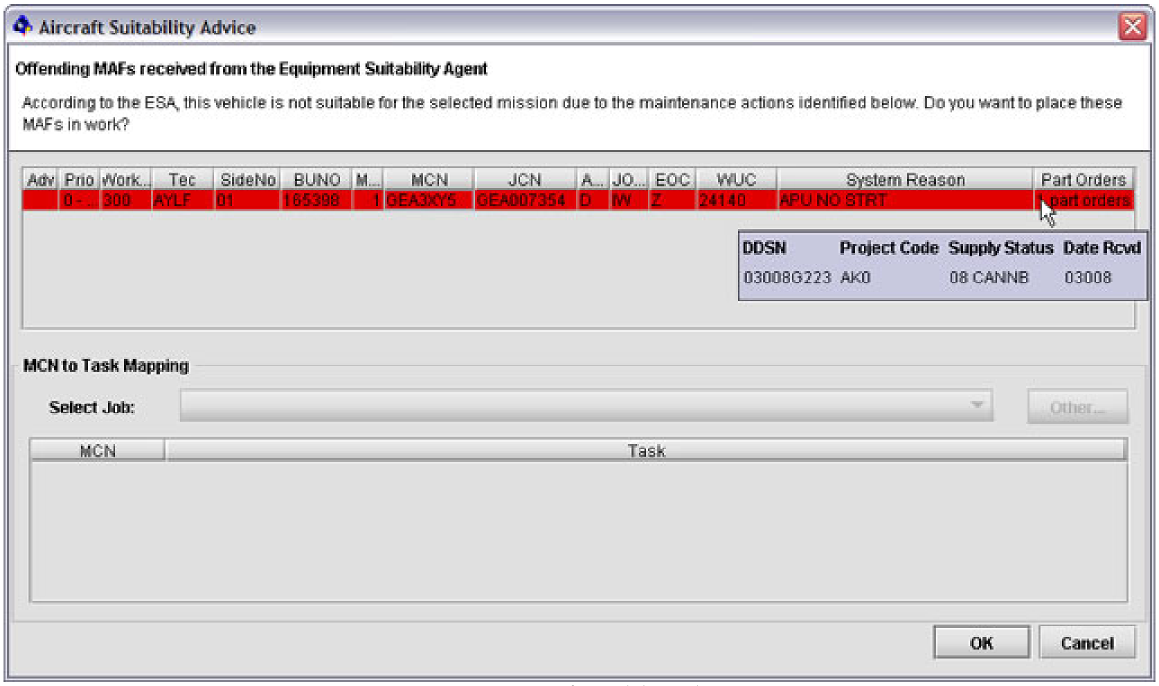


Figure 41. Aircraft Suitability Advice

The ASA agent, based on detailed information about the aircraft's status and personality and on the specific requirements of the individual mission profile, will identify any outstanding maintenance issues that should be resolved before the mission in question will take place.

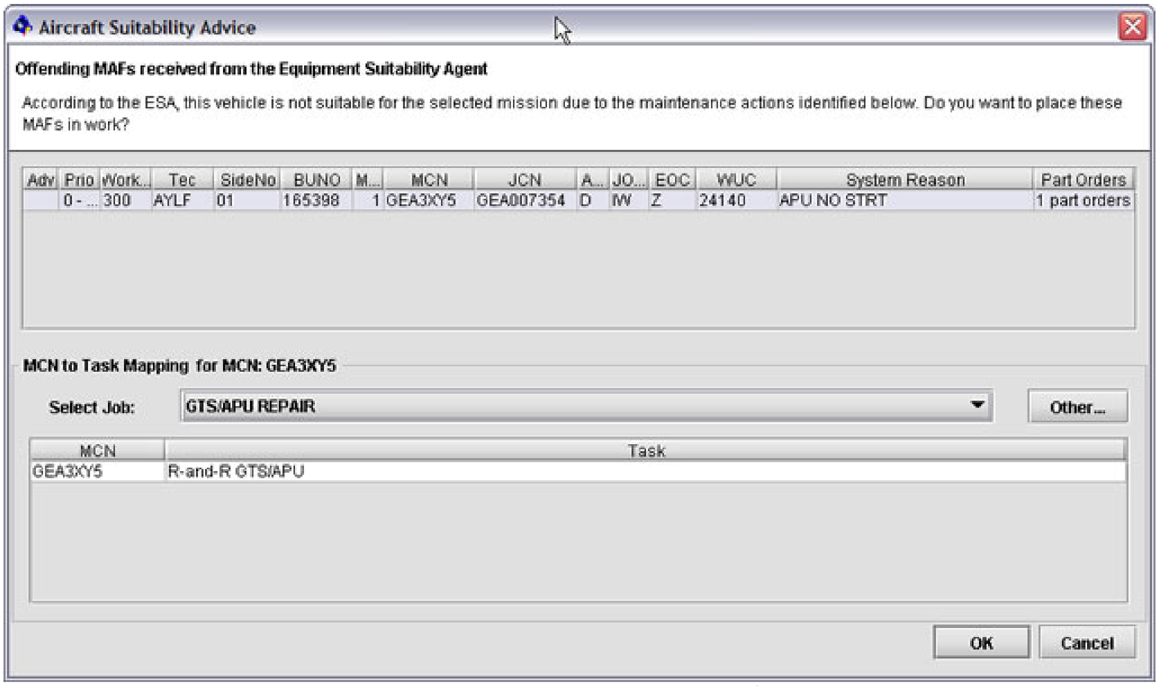


Figure 42. Mapping MAFs to Repair Manuals

Selecting a row in the top table will reveal how MAPLANT intends to map the specified Job Control Number (JCN) and its associated Maintenance Action Forms (MAFs) to individual repair actions defined in MAPLANT's repair manuals. In the example above, there is only one MAF (colored red in the previous screen shot), and selecting it reveals that MAPLANT has recognized this translates into a *"GTS/APU REPAIR"* type of repair. The resource and timing requirements of a *GTS/APU REPAIR* job are defined in MAPLANT's repair manuals. Note, you can examine the details of the repair manual using the appropriate menu item under the *Tools* menu.

If the user presses *Cancel* at this point, the manual aircraft assignment will be cancelled. If, on the other hand, the user presses the *OK* button, the assignment will be forced in and the associated repair(s) will be elevated in priority as recommended by the ESA agent.

If there are any outstanding part orders for the associated job, the user will be forced to enter estimated delivery dates, which will eventually be used as the *release time* of the repair action to be scheduled.

Advanced Topics
Violations and Alerts

Scheduling Violations

In the event that MAPLANT's solvers can not compute perfect solutions, it will resort to the next best solution it can find. This means that there may be aspects of the plan that are non-optimal. If so, any such problems will be pinpointed on the *Alerts Board*. An example is shown below.

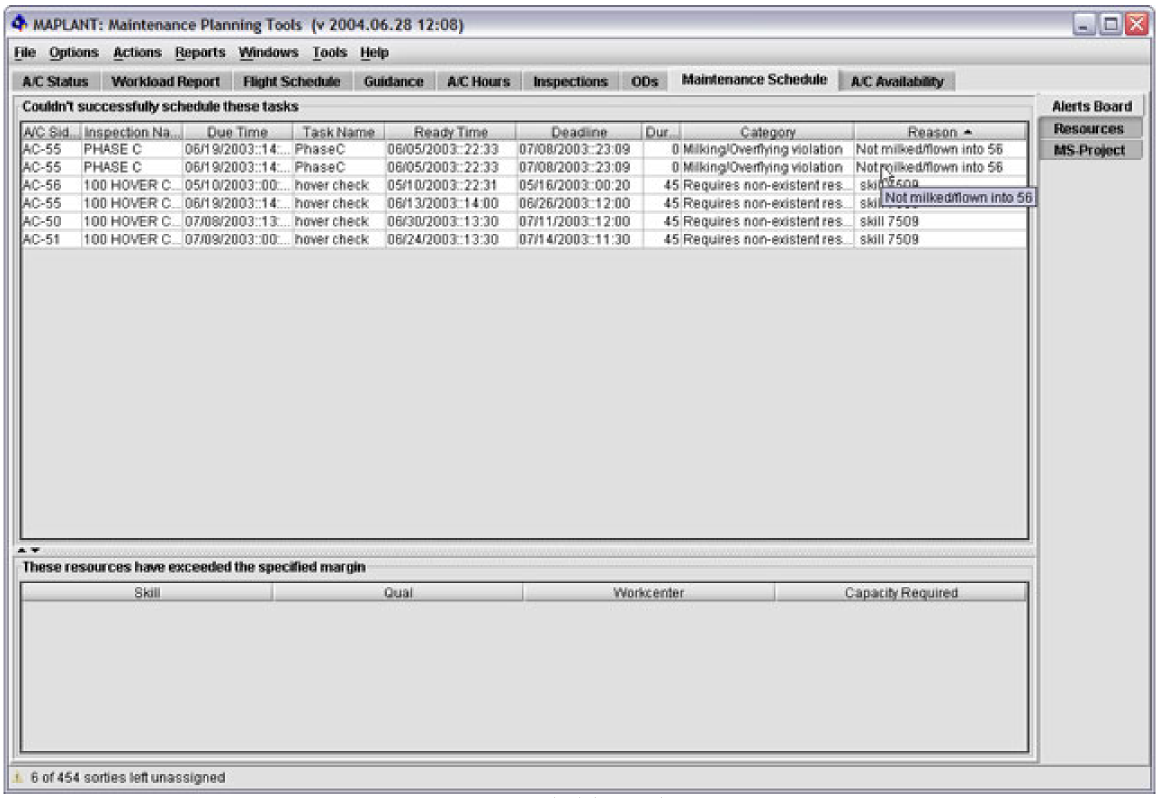


Figure 43. Scheduling Violations

In this example, the system warns the user of two general categories of violations. The last four violations describe problems with scheduling four different *100-Hour Hover Check* inspections. This is due to the fact that, in this instance, MAPLANT was specifically configured not to actively manage the allocation of pilots, and pilots are the only resources qualified to perform a hover check. Thus, in this case, maintenance will have to coordinate with operations to arrange for test pilots to be available during the required time periods.

The second class of violations above relate to MAPLANT not being able to schedule an aircraft's *Phase Inspection* at the same time as its upcoming *56-Day Special Inspection*.

Performing these two inspections simultaneously is one of the domain-specific *tricks of the trade (*due to efficiency reasons) therefore MAPLANT will attempt to do so when the situation allows. However, this is a soft constraint or a *preference* that is not considered an error with respect to the schedule being valid.

Milking Violations

As discussed in the previous section, MAPLANT will schedule certain inspections at the same time whenever possible. If the due windows of such inspections just miss each other, however, it may be possible to manipulate the timing of the due points by providing specific aircraft utilization guidance. The following screen shows an example where AC-55's phase inspection (blue) must be finished a few days before the 56-day inspection (black) may begin execution.

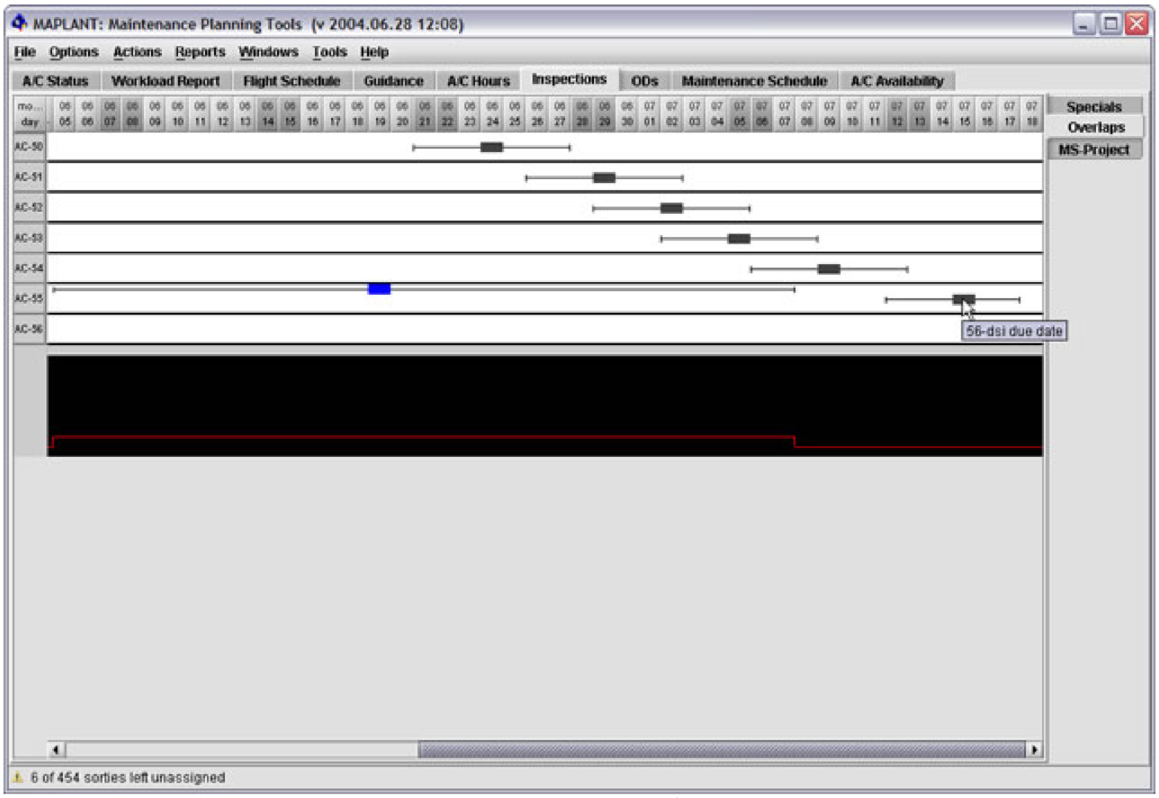


Figure 44. Inspection Overlaps

For usage-based inspections, we can extend the window of an inspection by reducing the amount of usage on the aircraft. Similarly the ready point of an inspection's due window may be moved forward in time by flying the aircraft more than normal. In the example above, we would like to fly AC-55 slightly less than its average share of the flight load. But how many fewer flight hours are actually needed? Below, a flight hour target icon has been added to the overlaps screen by right-clicking in the area of the row corresponding to the aircraft's phase inspection.

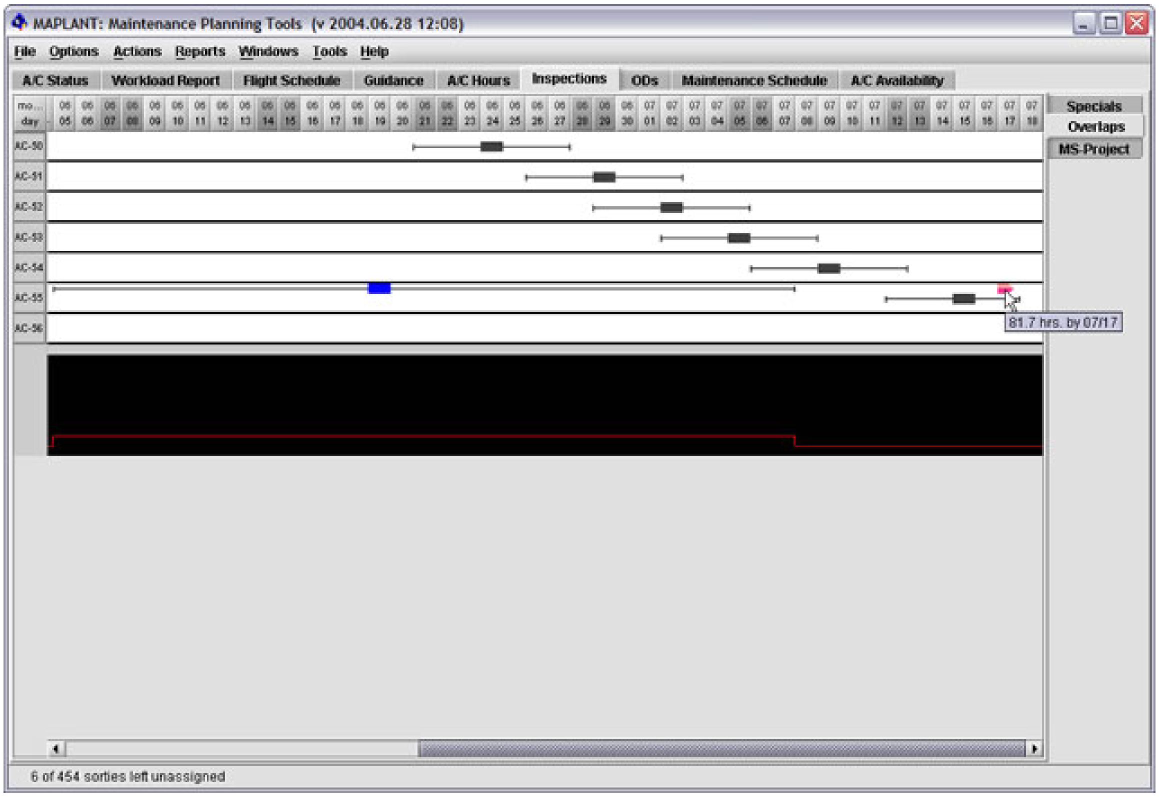


Figure 45. Milking Target

The resulting mouse-over indicates that the phase inspection will drop dead when the aircraft receives an additional 81.7 flight hours. Using the default aircraft utilization guidance, this plane will accumulate those 81.7 hours on July the 7th (07/07). In order to increase the chance that the scheduling system can overlap the phase and 56-dsi, however, we would prefer to accumulate the additional 81.7 hours of flight on or about July 17th (07/17).

Therefore, it would be desirable to manipulate the aircraft utilization guidance.

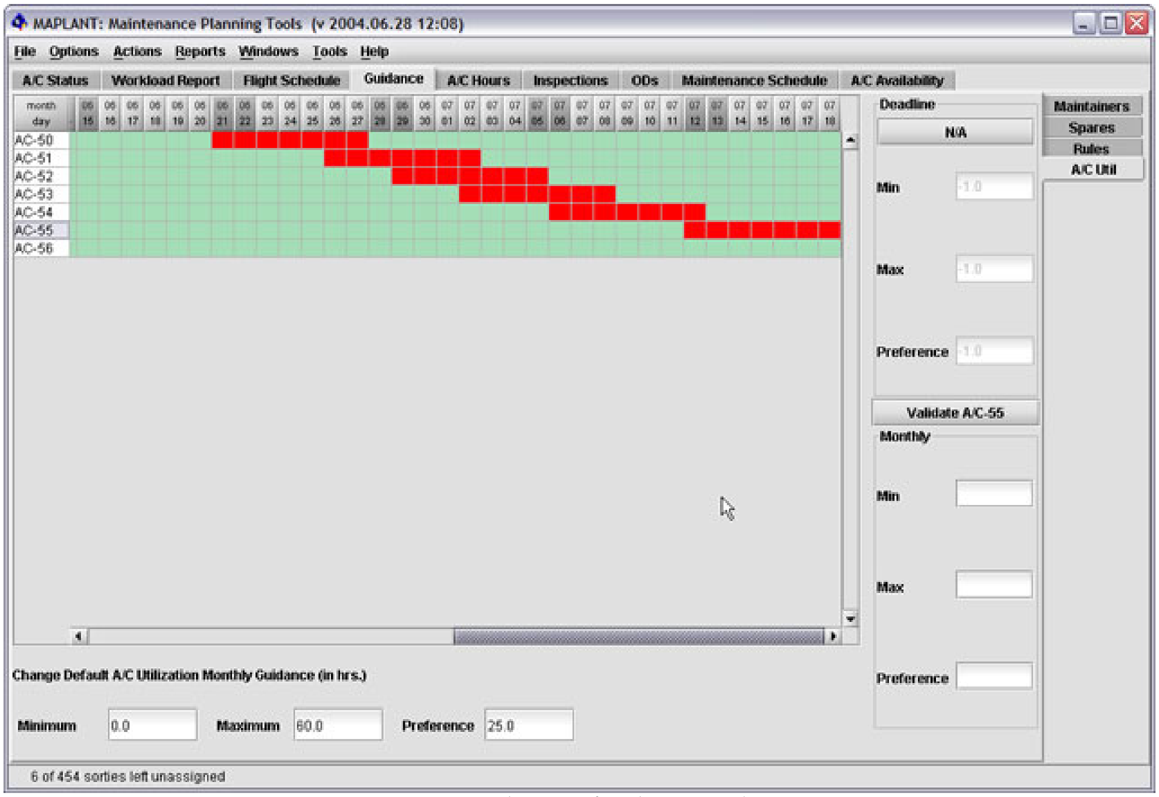


Figure 46. Initialize Aircraft Utilization Guidance

Comparing the default utilization guidance for AC-55 above with the modified guidance below, you can see that a *Deadline Preference* was added.

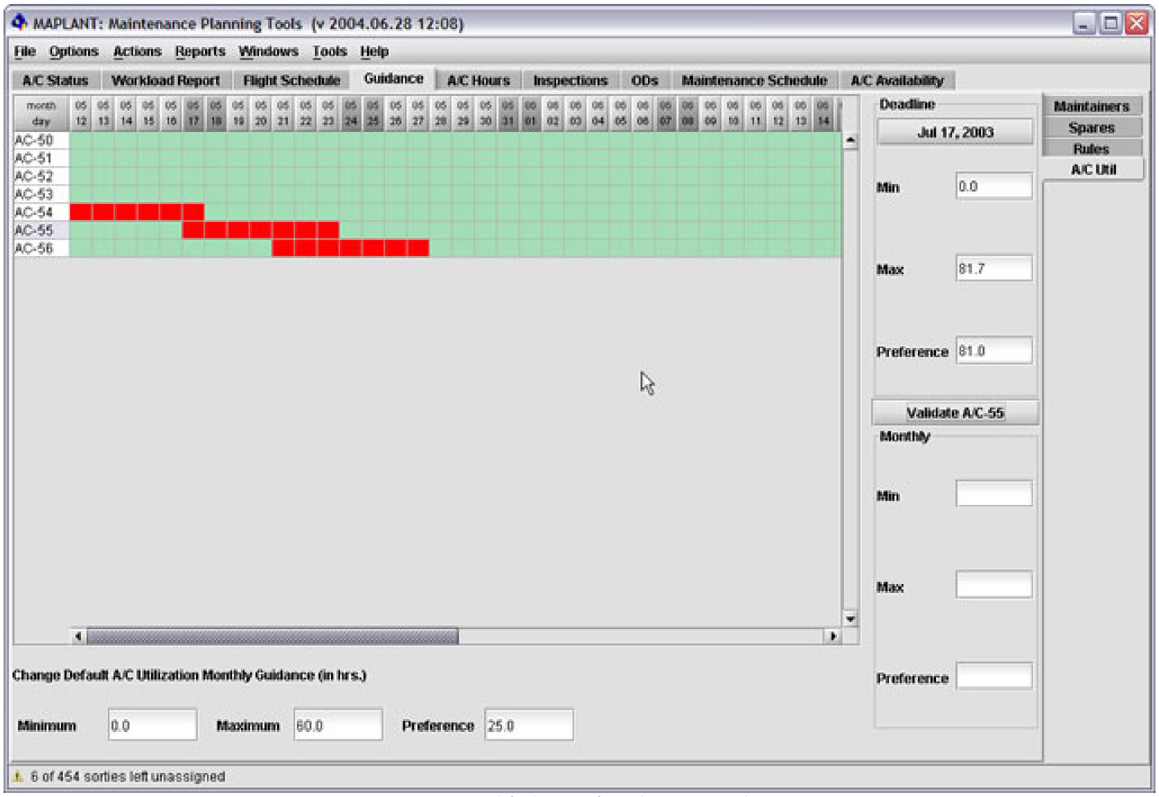


Figure 47. Modified Aircraft Utilization Guidance

This guidance informs the system to attempt to assign no more than 81.7 flight hours to AC-55 by July 17th and to assign the aircraft to sorties in a manner that gets as close as possible to 81.0 additional flight hours by the same date.

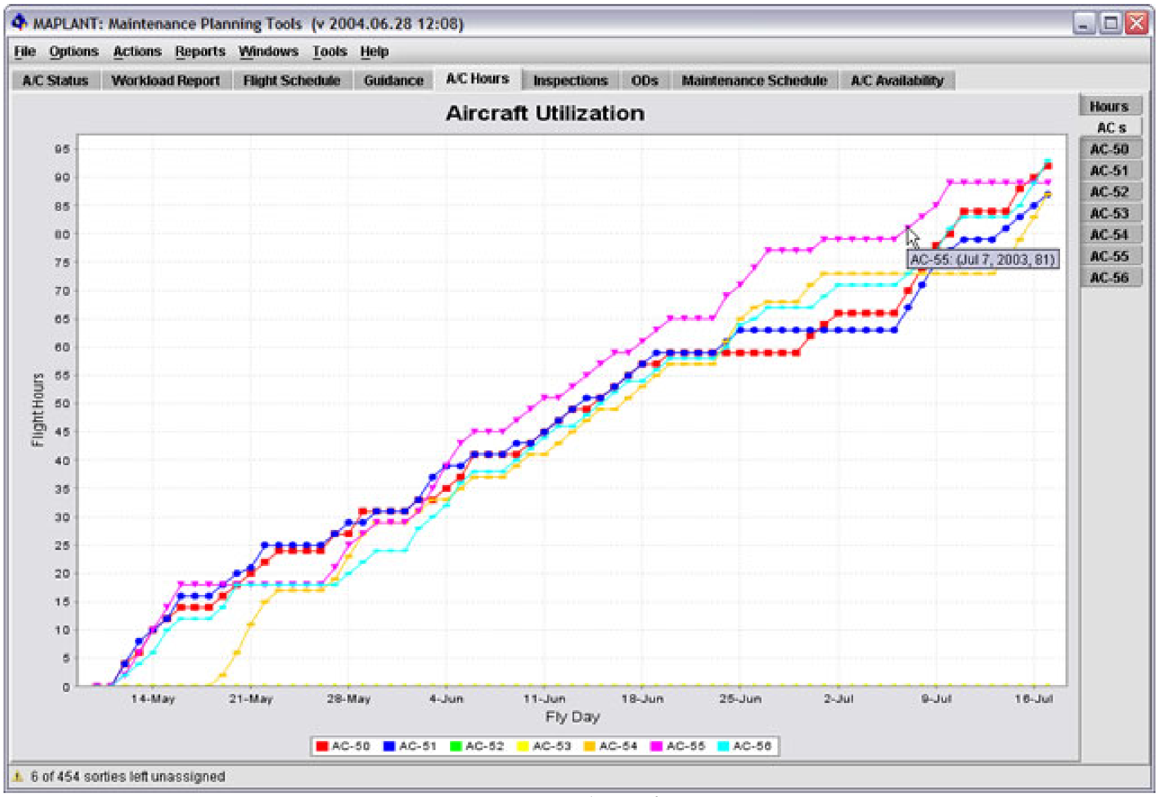


Figure 48. Original Aircraft Hours

After rerunning the aircraft assignment solver (i.e. see the *Actions* menu), the flight hour distribution across the fleet should change from the default rule of equitably assigning events to the aircraft to *milking* or holding AC-55 back slightly. Comparing the before (above) and after (below) views shows that AC-55 now flies a few fewer sorties, while the other aircraft take up the slack where possible.

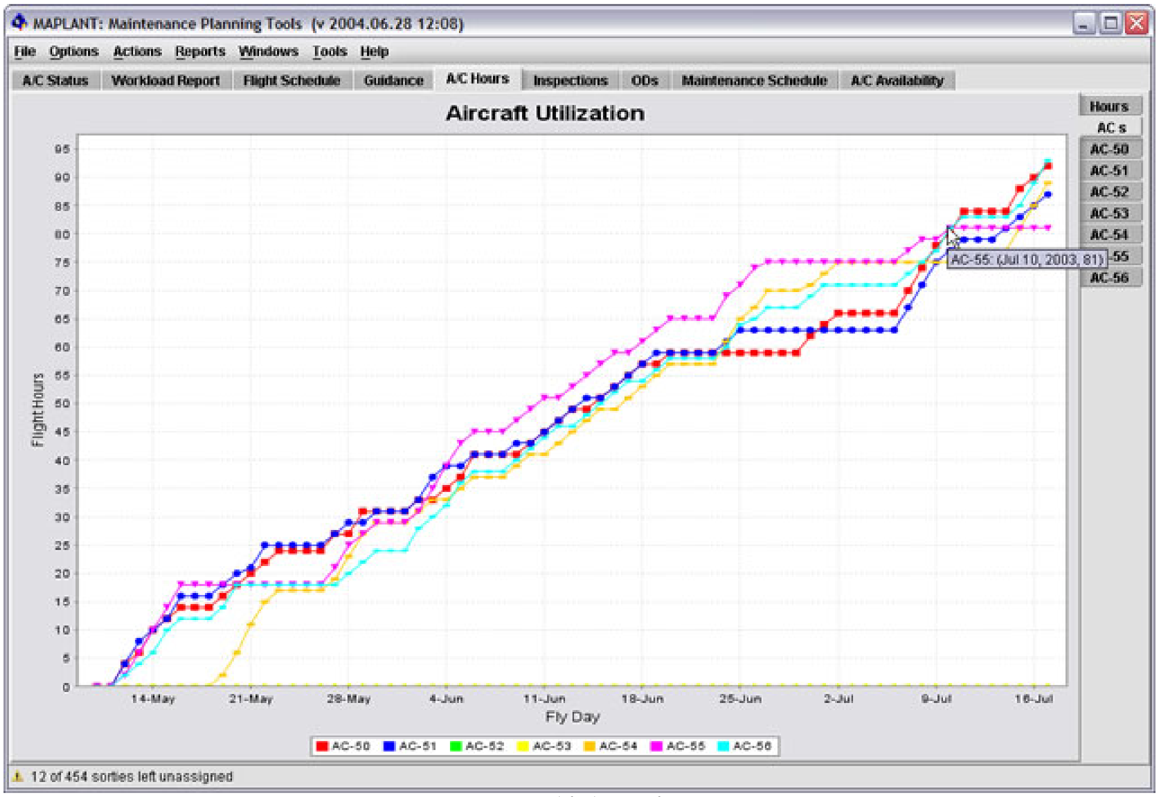


Figure 49. Modified Aircraft Hours

Finally, note that the phase and 56 due windows overlap now.

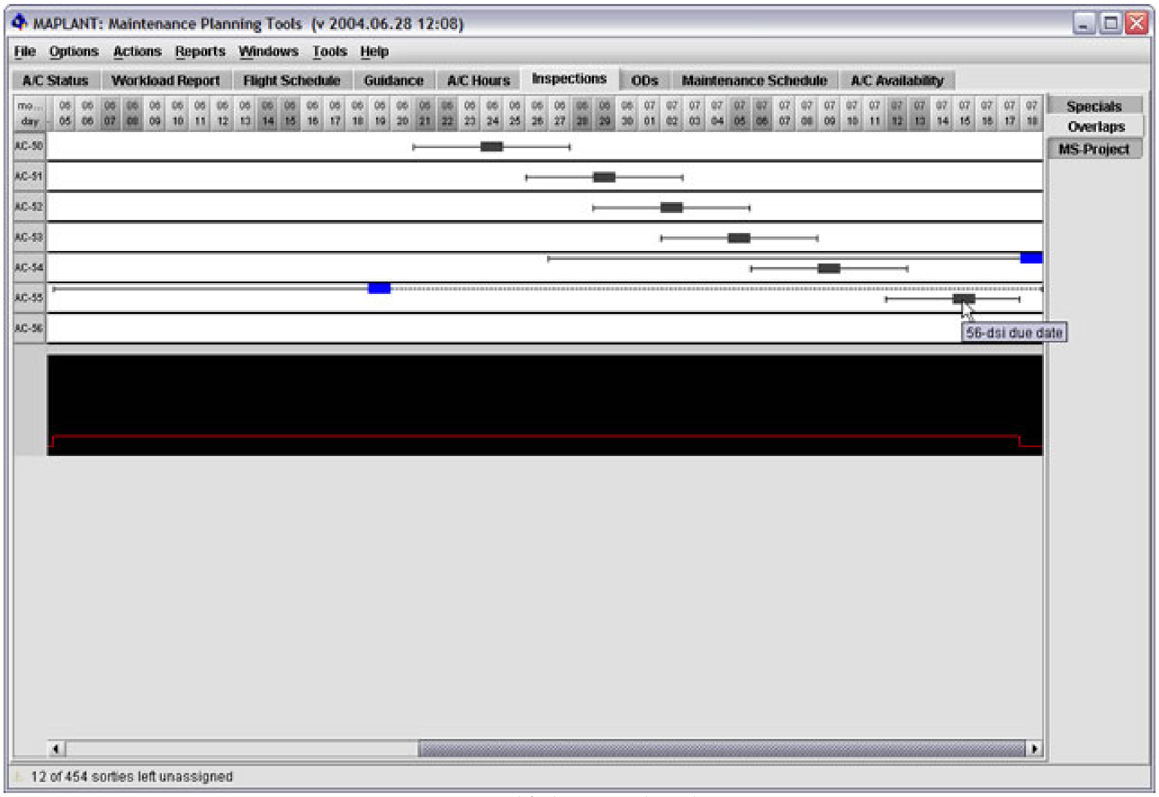


Figure 50. Modified Due Window Placement

And after rescheduling the maintenance, the milking violations disappear.

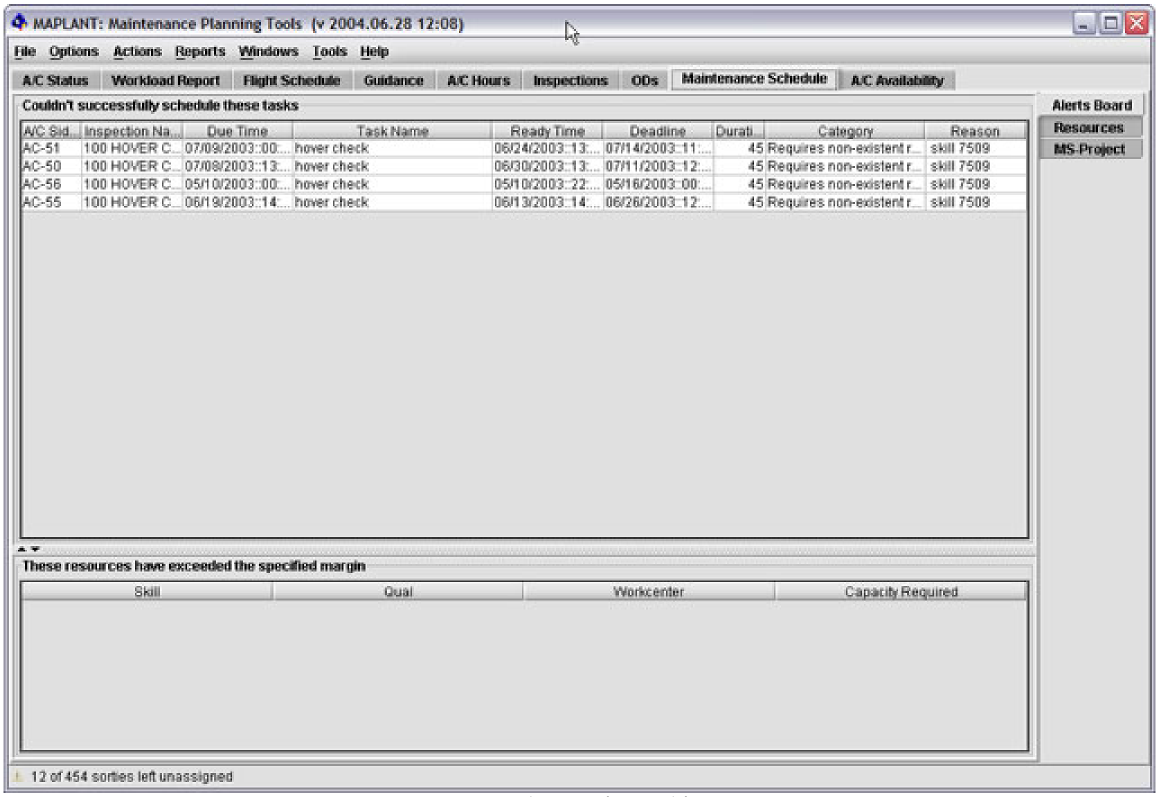


Figure 51. Violations After Modifications

Appendix 2

Summary PowerPoint slides that present the MAPLANT project. This presentation includes some of the figures included in this report.

MICANTS

Model-Integrated Computing and Autonomous Negotiation Teams for Autonomic Logistics

Gabor Karsai
Vanderbilt University/ISIS
J. Doyle, B. Laddaga, MIT
Boeing
R. Currer, IDEA Services

1

Administrative

- **Project Title:** MICANTS
- **Program Manager:** Vijay Raghavan
- **PI Name(s):** G. Karsai, H. Shrobe (MIT), M. Smith (Boeing), R. Currer (IDEA)
- **PI Phone Number(s):** 615 343 7471
- **PI E-Mail Address(es):** gabor.karsai@vanderbilt.edu
- **Company/Institution:** Vanderbilt University/ISIS
- **Contract Number:** F30602-99-2-0505
- **AO Number:** J132
- **Award Start Date:** 5/99
- **Award End Date:** 05/04 (FY04)
- **Agent Name/Organization:** R Hyle, AFRL, Rome/NY

2

Subcontractors and Collaborators

- Subcontractors
  - MIT:
    - Negotiation algorithms, preferences
  - The Boeing Company:
    - Domain scenarios
  - IDEA Services:
    - Domain expertise, USMC customer support
- Collaborators
  - USC/ISI:
    - Problem domain, system-to-system negotiation
  - Altarum
    - Improvements in solver
  - Washington University
    - Improvements in solver

3

Problem Description/Objective

- How to use
  1. Model-Integrated Computing, and
  2. Agent/Negotiation technology

  to solve complex resource management problems in (Autonomic) Logistics
- To demonstrate the feasibility of the technology through real-life example(s)

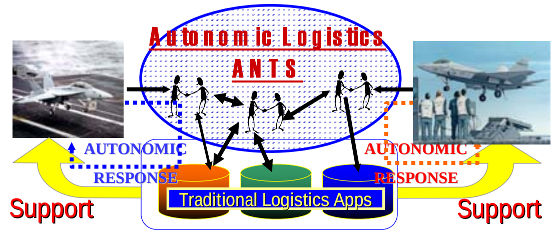
Contribution to ANTS

- Technology demonstration vehicle for negotiation
- Opportunity for working with real-life problems and data sets
- Demonstrating utility in a DoD context

Website: http://www.isis.vanderbilt.edu/Projects/micants/micants.htm
Demo: http://www.isis.vanderbilt.edu/Projects/micants/maplant/index.html

4

Model-Integrated Computing and Autonomous Negotiating Teams for Autonomic Logistics

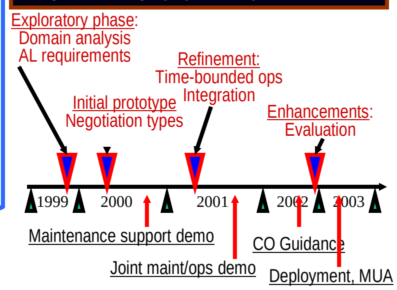


NEW IDEAS

- Modeling enables the analysis and synthesis of ANTs and system interfaces
- Dynamically evolving, negotiation-dependent preferences and value tradeoffs
- Rapid, incremental, and time-bounded negotiation processes
- Negotiation through incremental satisfaction of distributed constraints
- Autonomous logistics realized through integration of legacy systems by ANTs

IMPACT

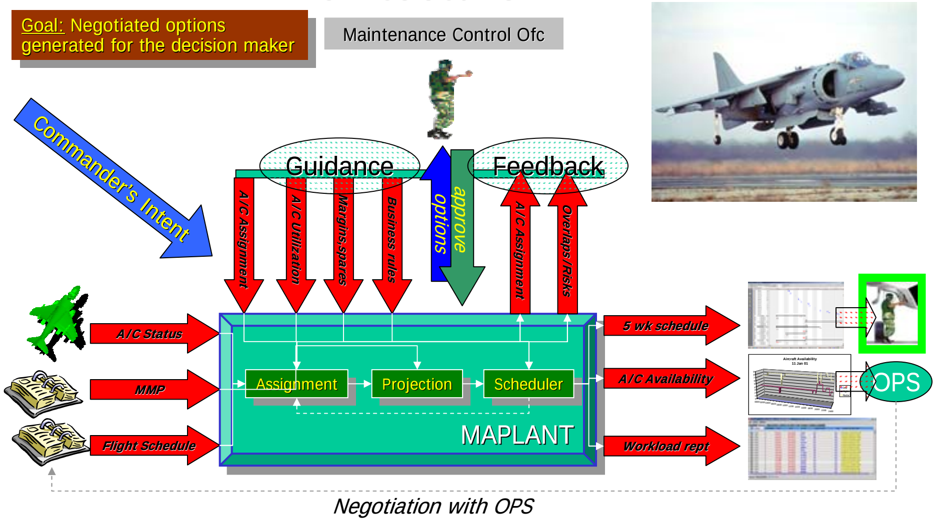
- Improvements in efficiency of current maintenance logistics processes
  - Increased combat readiness and efficiency
  - Decrease in cost
  - Reduced accident rate
- Intelligible automated processes that support distributed human decision-making save work
- Flexible approach that permits easy and rapid customization and adaptation to dynamic environments
- Existing systems are integrated into a cooperative environment



IDEA Services

Project Status

MAPLANT - Architecture



6

Project Description and Current Status

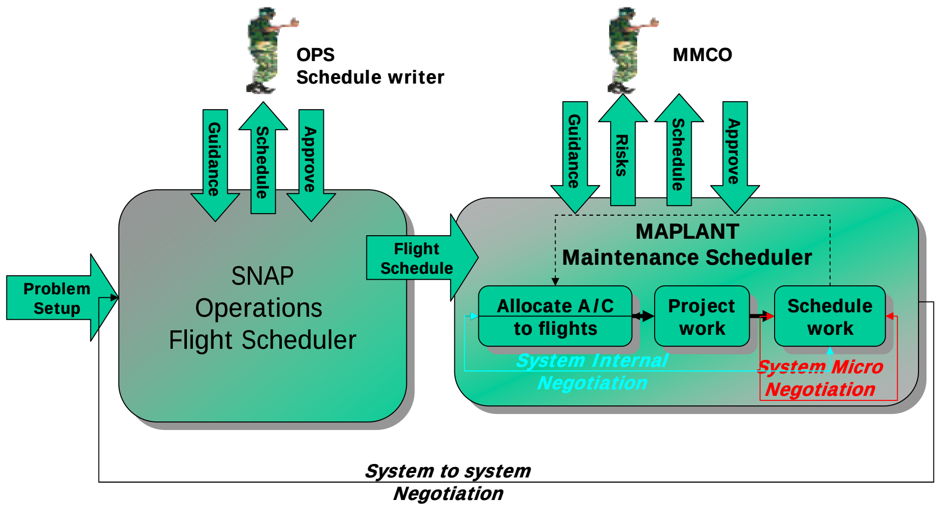
The overall technical approach:

- Negotiation technology (NT) is a small, yet crucial ingredient in complex decision support systems. NT applications can be built using a number of different technologies. We have used 3 different approaches:
  1. On the *system-to-system* level: a messaging protocol to interchange (domain-specific) solutions/requirements and resource availabilities
  2. On the *system internal* level: a messaging protocol to interchange schedules and change requests on those schedules
  3. On the *system micro* level: fine-grain constraint propagation

7

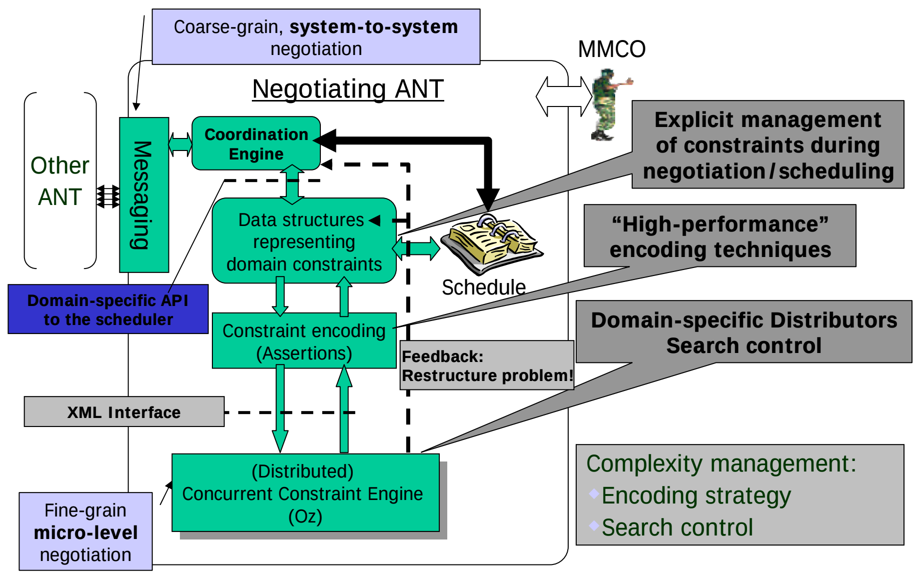
Technical Approach: Negotiation

Within/across the systems



8

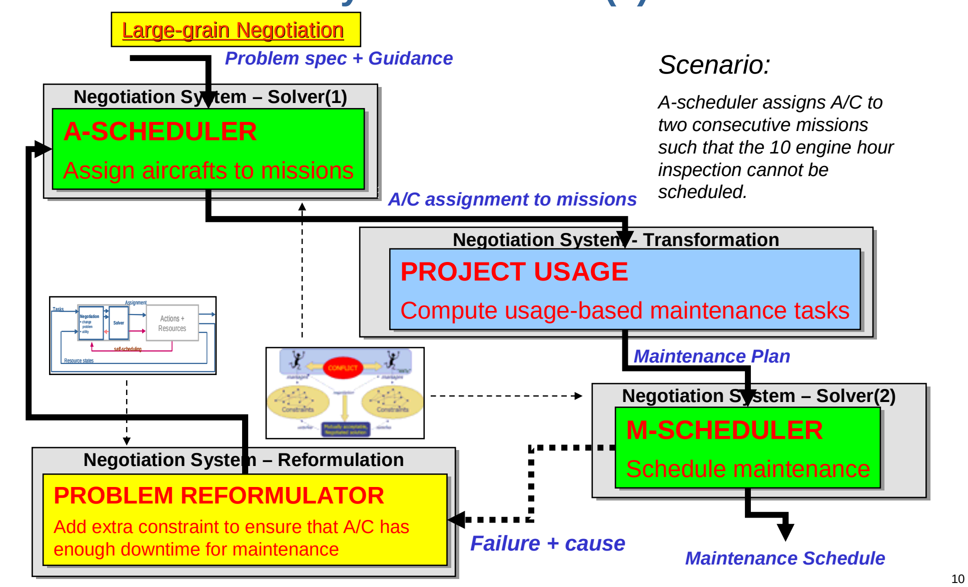
Technical Approach: Negotiation



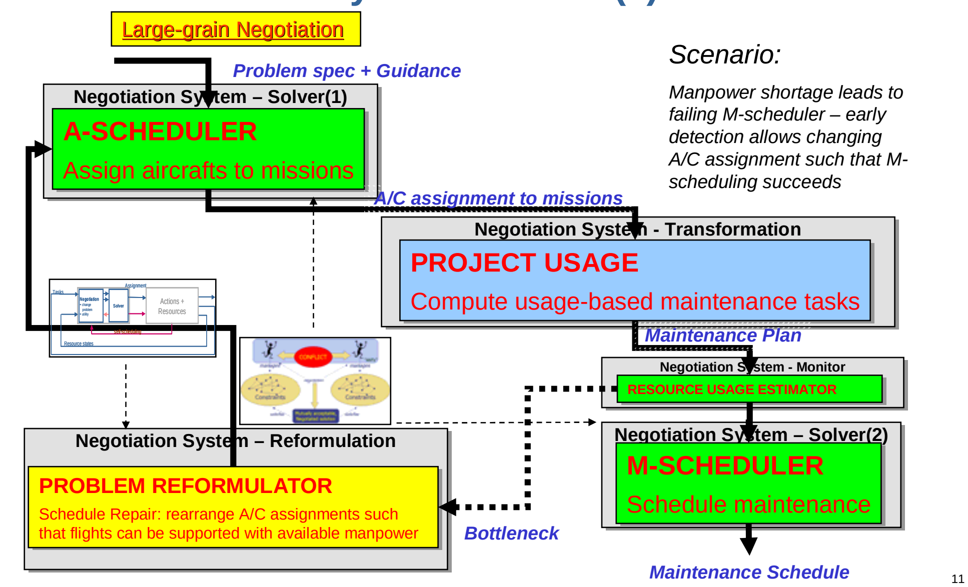
9

Technical Approach: Negotiation

System Internal (1)



Technical Approach: Negotiation
System Internal (2)



Technical Approach: Negotiation

Micro-level negotiation: CCP (1)

MAPLANT Scheduler Problem Solving paradigm

Problem: Expressed as constraints posted to a constraint store

- Primitive constraints: single variable domain restriction
- Complex constraints: multi-variable expressions, pending "threads"
- When posted, contradictions are detected immediately

Negotiation: (Within scheduler)

Constraint solving process (a *complete* solution method)

- *Propagate* constraints: Domain restriction
  - Run threads until "space" stabilizes – "**Negotiate over variable domains"**
- *Pick* a constraint C and *distribute over* C and ~C: Split in search tree
  - Example: C => "X:=N" – **Pick a partial solution and compute its consequences**
  - **"Take a negotiation stance and check it against other constraints"**

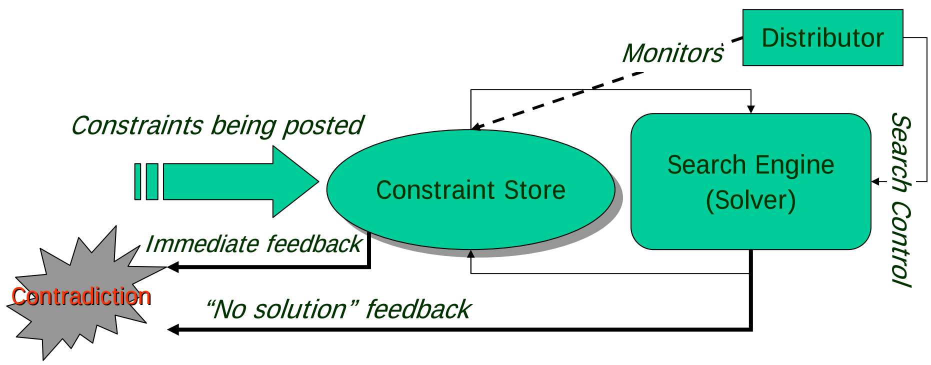
Controlling search (-> "Computational Complexity")

- Monitor constraint postings
- Distribute according to "situation" in constraint store

12

Technical Approach: Negotiation

Micro-level negotiation: CCP (2)



Guidance / Soft constraints:
- **Can be “negotiated away”**

“Laws” / Hard constraints:
- **Non-negotiable**

Results:

- **Size: 3,000+ tasks, 200+ maintainers, 80% margins**
- **~10K complex constraints**
- **Planning window: 5 weeks, 15 minutes resolution**
- **Scheduled in ~70secs on P3/1.7GHz/512MB laptop**

13

Major accomplishments

- Militarily relevant, working decision support application that is used by real users and is being transitioned to the services.
- Reusable technology that can be applied to other domains as well
  - Scheduling of spacecraft launch operations (USAF 45th Space Wing)
  - Joint scheduling of operations and maintenance in USAF context (TBMS)
  - Scheduling maintenance for heterogeneous aircraft (CARTE)

14

Lessons learnt

Technical:

- There are many different ways of doing negotiation. Choose what works.
- Constraint programming (the way done in Oz) is a powerful programming paradigm.
- Problem encoding and clever solution techniques are key to high performance.
- The success of a decision support system depends on the ability to translate user requirements into constraints on the solution. These constraints govern the negotiation process.
- Explanations upon failure are needed ("What should I tweak?")

Logistics:

- User involvement is essential to the success of a project. Unused results are useless.
- To support tech transition, be flexible and make flexible tools.

15

Open Problems

Main aspects: Scaling, control, analyzability, explanation, proactive adaptation

1. Scaling to larger organizations and horizons
2. Tools ("language") for specifying constraints, preferences, intent, and guidance
3. Languages for coding solvers: distributed search engines/reformulators
4. Ensuring stability and convergence in asynchronous negotiation protocols
5. Techniques for compiling constraints/preferences into "code" used in the search
6. Early prediction/estimation of problem difficulty and feedback to negotiation
7. Explanation structures (when the solver fails): fine-grain analysis
8. Handling priorities, tradeoffs and language constructs to express these mechanisms
9. Rapidly computing many, alternative solutions
10. Change propagation and incremental, minimum-disturbance recomputation of solutions
11. Coupling execution monitoring to trigger re-negotiation
12. Coordinated planning (as opposed to scheduling) through negotiation
13. Integration with systems that provide probabilistic assessments (e.g. PHM)
14. Negotiation between planning and scheduling
15. Goal-driven negotiation to achieve an outcome

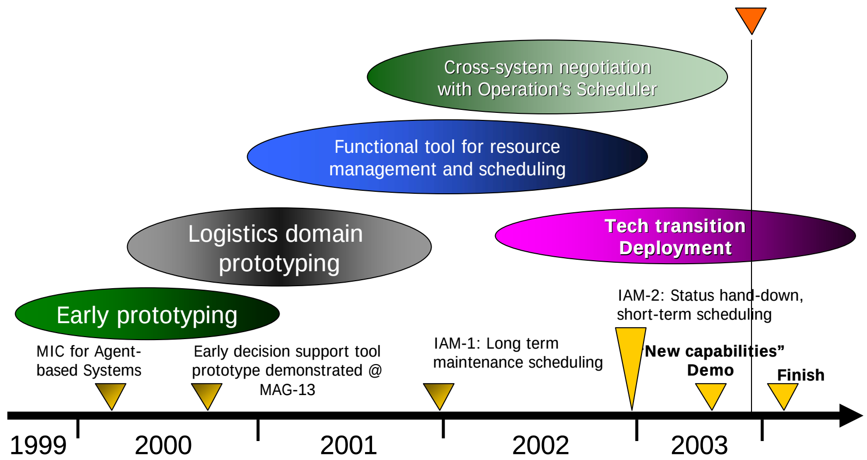
16

Deliverables

- MAPLANT Tool:
  - http://www.isis.vanderbilt.edu/Projects/micants/maplant/index.html
- Key publications:
  - van Buskirk C., Dawant B., Karsai G., Sprinkle J., Szokoli G., Suwanmongkol K., Currer,Russ: Computer-aided Aircraft Maintenance Scheduling, ISIS-02-303, November, 2002.
  - Sprinkle J., van Buskirk C., Karsai G.: Analysis and Representation of Clauses in Satisfiability of Constraints, ISIS-01-205, August 6, 2001.
  - Sprinkle J., van Buskirk C., Karsai G.: Modeling Agent Negotiation, IEEE International Conference on Systems, Man, and Cybernetics, Proceedings, Nashville, TN, October 8, 2000.
  - Karsai G., Bloor G., Doyle J.: Automating Human Based Negotiation Processes for Autonomic Logistics, Proceedings of the IEEE Aerospace 2000, CD-ROM Reference 11.0302, Big Sky, MT, March, 2000.

17

Project Schedule and Milestones



18

Technology Transition/Transfer

- CACE ACTD – MAG-13 - USMC
  - Prototype installed @ MAG-13 (4 squadrons), LHA for deployment (2 squadron-)
  - MUA (CACE ACTD)
  - USMC Transition Manager named
- LM/JSF
  - TTA signed ; Integrated demo (w LM ALIS) Jan/Feb 04
- Other targets:
  - CARTE (ONR FNCS – continuing research)
    - Extensions to other, heterogeneous systems
  - USAF 45th Space Wing
    - Technical Interchange Meeting in early Oct ; Plan for building a feasibility demo
  - USAF MITWG
    - “Coordinated operations and maintenance scheduling is a requirement”
  - J-UCAS (Boeing)
    - Early evaluation of MAPLANT
  - USAF – HESR: LOCIS, OMDS (Kelley Log)
    - Maintenance scheduling prototype using MAPLANT’s solver
  - USN – Newport News Shipbuilding
    - Technical interchange meeting
  - USAF TBMS
    - Early discussions

Appendix 3

Summary PowerPoint slides that present the SPACEPLANT project. This presentation includes some of the figures included in this report.

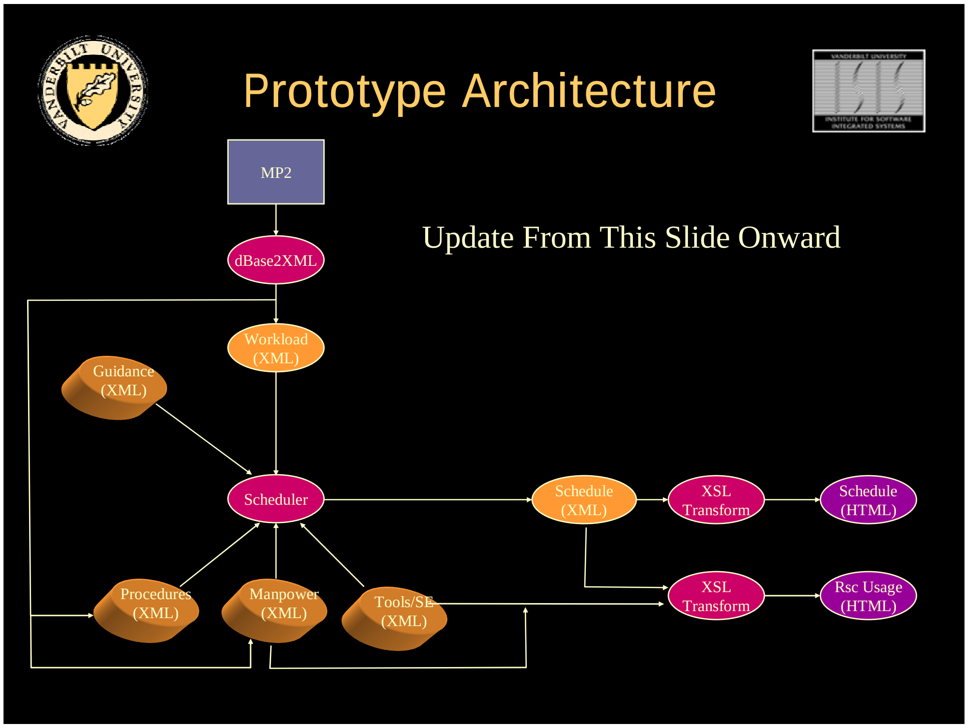
Institute for Software Integrated Systems
Vanderbilt University

SPACEPLANT

45th Space Wing Maintenance Scheduling Prototype

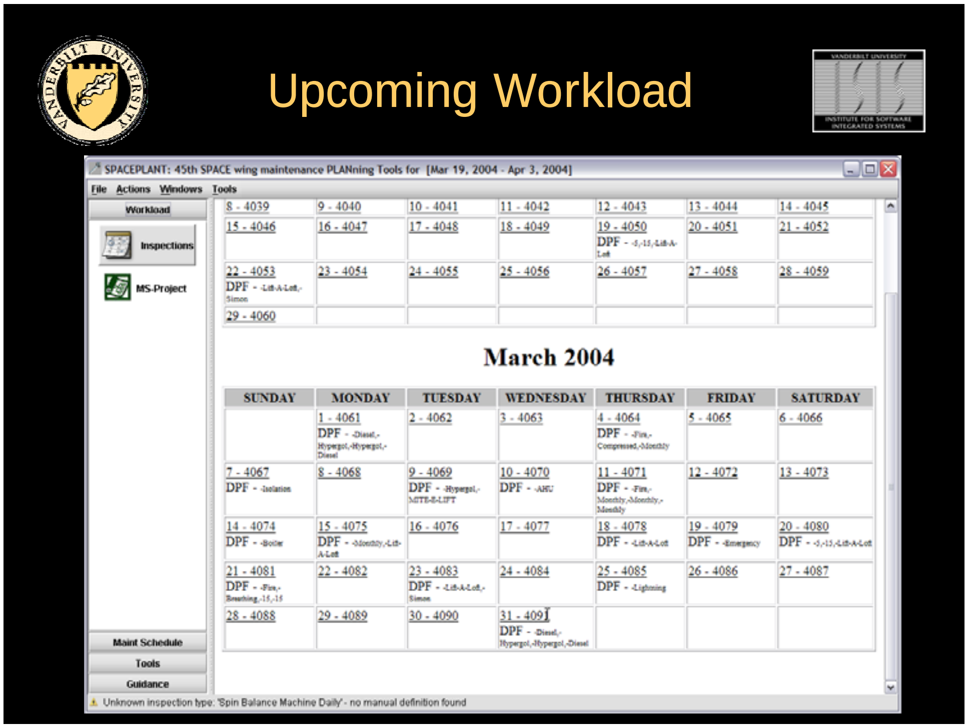
Project Overview

- The Maintenance Planning Tool (MAPLANT) has been developed in the framework of DARPA/IXO's Autonomous Negotiating Teams (ANTs) project. Currently, it provides decision support for assisting a Maintenance Control Officer (MCO) of an AV8-B squadron in (1) assigning air vehicles to missions, (2) scheduling maintenance activities to support those missions, and (3) interacting and negotiating with an automated flight scheduler tool to achieve coordination between the operations and the maintenance departments. This tool is using scheduling technology that potentially could be applicable in other contexts, namely scheduling problems arising in space system launch preparation. This project is an effort to explore potential benefits and to carry out a cursory identification of any requirements that would involve significant adaptation for the existing software to support this new domain.

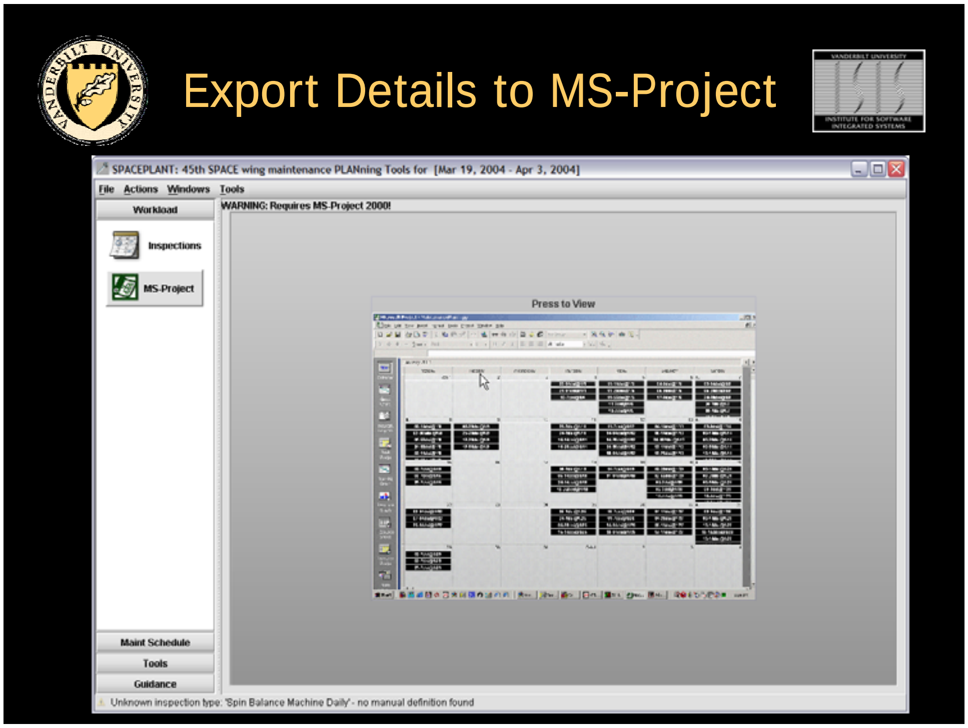


Prototype Architecture
MP2
dBase2XML
Update From This Slide Onward
Workload
(XML)
Guidance
(XML)
Scheduler
Schedule
(XML)
XSL
Transform
Schedule
(HTML)
Procedures
(XML)
Manpower
(XML)
Tools/SE
(XML)
XSL
Transform
Rsc Usage
(HTML)

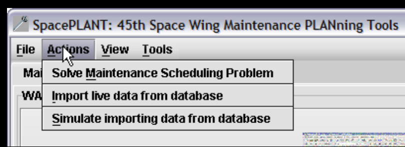
Screen Shots



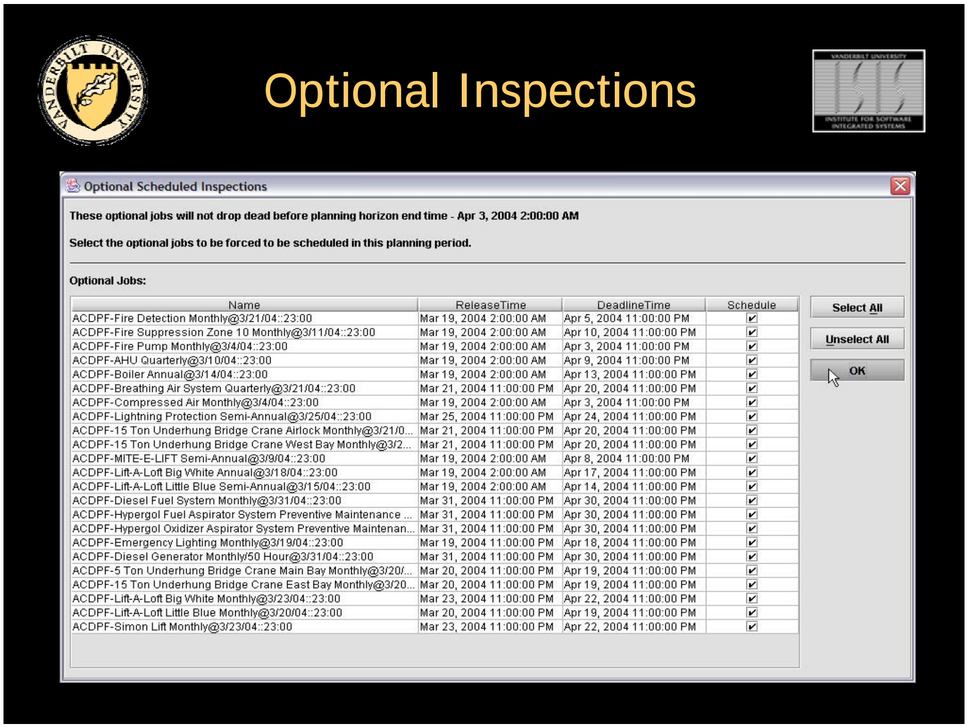
Upcoming Workload
SPACEPLANT: 45th SPACE wing maintenance PLANning Tools for [Mar 19, 2004 - Apr 3, 2004]
File Actions Windows Tools
Workload
Inspections
MS-Project
8 - 4039
9 - 4040
10 - 4041
11 - 4042
12 - 4043
13 - 4044
14 - 4045
15 - 4046
16 - 4047
17 - 4048
18 - 4049
19 - 4050
20 - 4051
21 - 4052
22 - 4053
DPF - -Lift-A-Loft,-Simon
23 - 4054
24 - 4055
25 - 4056
26 - 4057
27 - 4058
28 - 4059
29 - 4060
March 2004
SUNDAY
MONDAY
TUESDAY
WEDNESDAY
THURSDAY
FRIDAY
SATURDAY
1 - 4061
DPF - -Diesel,-Hypergol,-Hypergol,-Diesel
2 - 4062
3 - 4063
4 - 4064
DPF - -Fire,-Compressed,-Monthly
5 - 4065
6 - 4066
7 - 4067
DPF - -Isolation
8 - 4068
9 - 4069
DPF - -Hypergol,-MITE-E-LIFT
10 - 4070
DPF - -AHU
11 - 4071
DPF - -Fire,-Monthly,-Monthly,-Monthly
12 - 4072
13 - 4073
14 - 4074
DPF - -Boiler
15 - 4075
DPF - -Monthly,-Lift-A-Loft
16 - 4076
17 - 4077
18 - 4078
DPF - -Lift-A-Loft
19 - 4079
DPF - -Emergency
20 - 4080
21 - 4081
22 - 4082
23 - 4083
DPF - -Lift-A-Loft,-Simon
24 - 4084
25 - 4085
DPF - -Lightning
26 - 4086
27 - 4087
28 - 4088
29 - 4089
30 - 4090
31 - 4091
DPF - -Diesel,-Hypergol,-Hypergol,-Diesel
Maint Schedule
Tools
Guidance
Unknown inspection type: 'Spin Balance Machine Daily' - no manual definition found


Export Details to MS-Project
VANDERBILT UNIVERSITY
INSTITUTE FOR SOFTWARE INTEGRATED SYSTEMS
SPACEPLANT: 45th SPACE wing maintenance PLANning Tools for [Mar 19, 2004 - Apr 3, 2004]
File Actions Windows Tools
Workload
WARNING: Requires MS-Project 2000!
Inspections
MS-Project
Press to View
Maint Schedule
Tools
Guidance
Unknown inspection type: 'Spin Balance Machine Daily' - no manual definition found

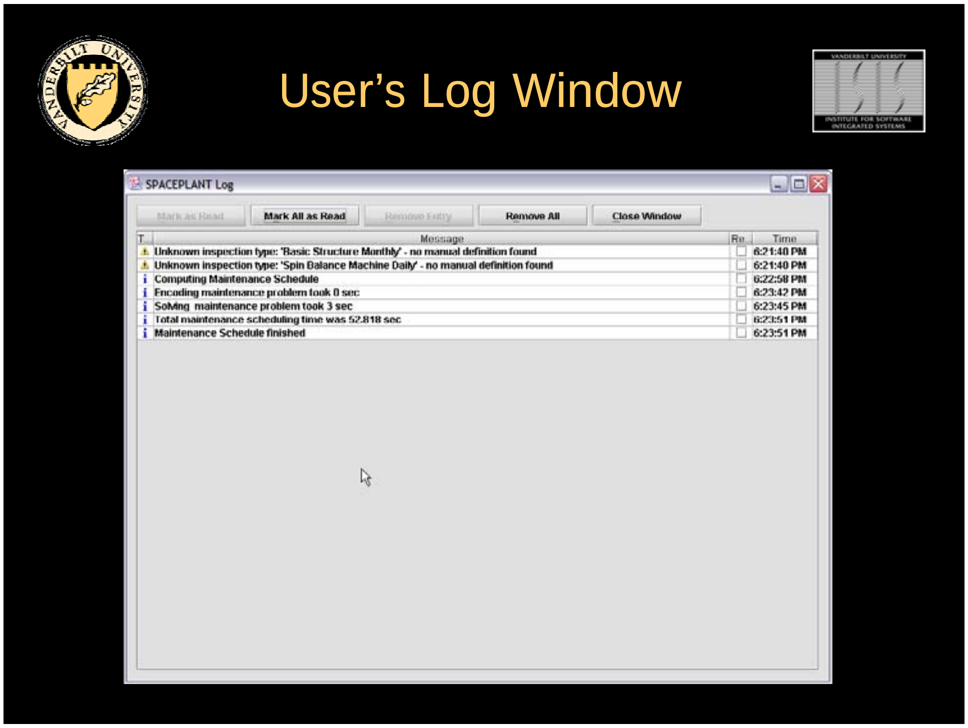
Actions Menu

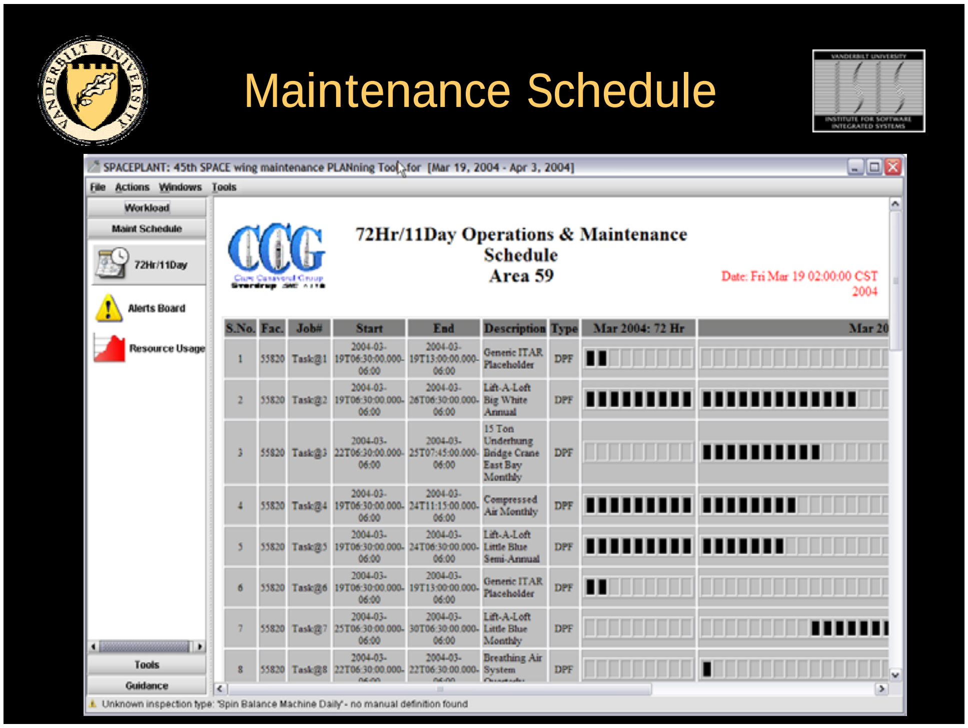


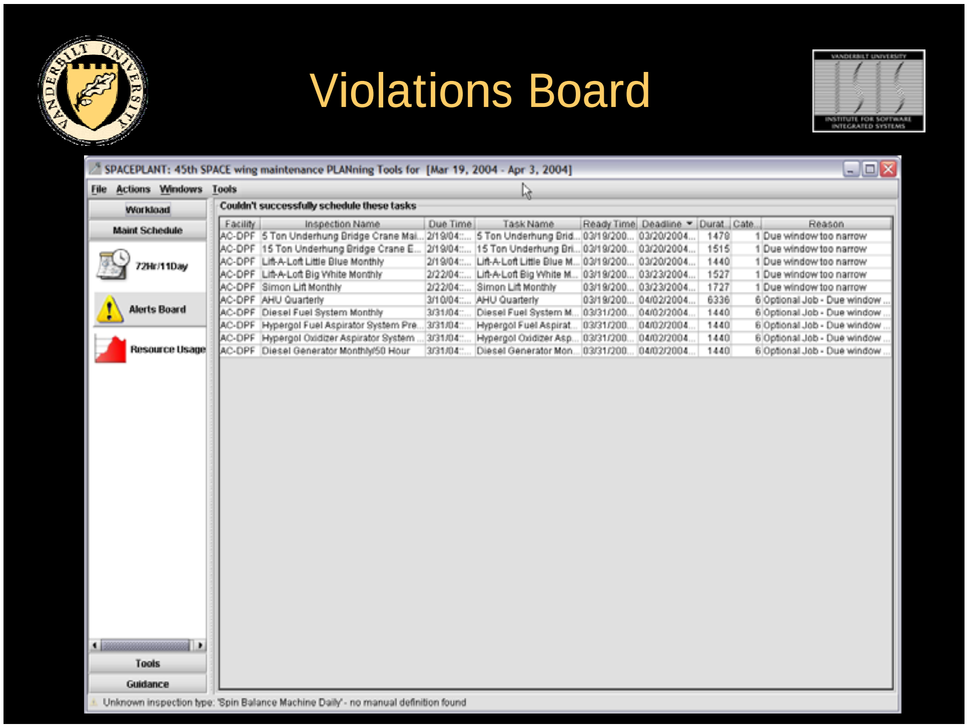
- Schedules the required work packages for the next 72/11 planning period.
- Imports data from a live database for a chosen point in time.
- Imports data for a particular planning period from off-line, static data (*for demonstration purposes*).



Name	ReleaseTime	DeadlineTime	Schedule
ACDPF-Fire Detection Monthly@3/21/04::23:00	Mar 19, 2004 2:00:00 AM	Apr 5, 2004 11:00:00 PM	✔
ACDPF-Fire Suppression Zone 10 Monthly@3/11/04::23:00	Mar 19, 2004 2:00:00 AM	Apr 10, 2004 11:00:00 PM	✔
ACDPF-Fire Pump Monthly@3/4/04::23:00	Mar 19, 2004 2:00:00 AM	Apr 3, 2004 11:00:00 PM	✔
ACDPF-AHU Quarterly@3/10/04::23:00	Mar 19, 2004 2:00:00 AM	Apr 9, 2004 11:00:00 PM	✔
ACDPF-Boiler Annual@3/14/04::23:00	Mar 19, 2004 2:00:00 AM	Apr 13, 2004 11:00:00 PM	✔
ACDPF-Breathing Air System Quarterly@3/21/04::23:00	Mar 21, 2004 11:00:00 PM	Apr 20, 2004 11:00:00 PM	✔
ACDPF-Compressed Air Monthly@3/4/04::23:00	Mar 19, 2004 2:00:00 AM	Apr 3, 2004 11:00:00 PM	✔
ACDPF-Lightning Protection Semi-Annual@3/25/04::23:00	Mar 25, 2004 11:00:00 PM	Apr 24, 2004 11:00:00 PM	✔
ACDPF-15 Ton Underhung Bridge Crane Airlock Monthly@3/21/0...	Mar 21, 2004 11:00:00 PM	Apr 20, 2004 11:00:00 PM	✔
ACDPF-15 Ton Underhung Bridge Crane West Bay Monthly@3/2...	Mar 21, 2004 11:00:00 PM	Apr 20, 2004 11:00:00 PM	✔
ACDPF-MITE-E-LIFT Semi-Annual@3/9/04::23:00	Mar 19, 2004 2:00:00 AM	Apr 8, 2004 11:00:00 PM	✔
ACDPF-Lift-A-Loft Big White Annual@3/18/04::23:00	Mar 19, 2004 2:00:00 AM	Apr 17, 2004 11:00:00 PM	✔
ACDPF-Lift-A-Loft Little Blue Semi-Annual@3/15/04::23:00	Mar 19, 2004 2:00:00 AM	Apr 14, 2004 11:00:00 PM	✔
ACDPF-Diesel Fuel System Monthly@3/31/04::23:00	Mar 31, 2004 11:00:00 PM	Apr 30, 2004 11:00:00 PM	✔
ACDPF-Hypergol Fuel Aspirator System Preventive Maintenance ...	Mar 31, 2004 11:00:00 PM	Apr 30, 2004 11:00:00 PM	✔
ACDPF-Hypergol Oxidizer Aspirator System Preventive Maintenan...	Mar 31, 2004 11:00:00 PM	Apr 30, 2004 11:00:00 PM	✔
ACDPF-Emergency Lighting Monthly@3/19/04::23:00	Mar 19, 2004 11:00:00 PM	Apr 18, 2004 11:00:00 PM	✔
ACDPF-Diesel Generator Monthly/50 Hour@3/31/04::23:00	Mar 31, 2004 11:00:00 PM	Apr 30, 2004 11:00:00 PM	✔
ACDPF-5 Ton Underhung Bridge Crane Main Bay Monthly@3/20/...	Mar 20, 2004 11:00:00 PM	Apr 19, 2004 11:00:00 PM	✔
ACDPF-15 Ton Underhung Bridge Crane East Bay Monthly@3/20...	Mar 20, 2004 11:00:00 PM	Apr 19, 2004 11:00:00 PM	✔
ACDPF-Lift-A-Loft Big White Monthly@3/23/04::23:00	Mar 23, 2004 11:00:00 PM	Apr 22, 2004 11:00:00 PM	✔
ACDPF-Lift-A-Loft Little Blue Monthly@3/20/04::23:00	Mar 20, 2004 11:00:00 PM	Apr 19, 2004 11:00:00 PM	✔
ACDPF-Simon Lift Monthly@3/23/04::23:00	Mar 23, 2004 11:00:00 PM	Apr 22, 2004 11:00:00 PM	✔


User's Log Window
SPACEPLANT Log
Mark as Read
Mark All as Read
Remove Entry
Remove All
Close Window
T
Message
Re
Time
Unknown inspection type: 'Basic Structure Monthly' - no manual definition found 6:21:40 PM
Unknown inspection type: 'Spin Balance Machine Daily' - no manual definition found 6:21:40 PM
Computing Maintenance Schedule 6:22:58 PM
Encoding maintenance problem took 0 sec 6:23:42 PM
Solving maintenance problem took 3 sec 6:23:45 PM
Total maintenance scheduling time was 52.818 sec 6:23:51 PM
Maintenance Schedule finished 6:23:51 PM

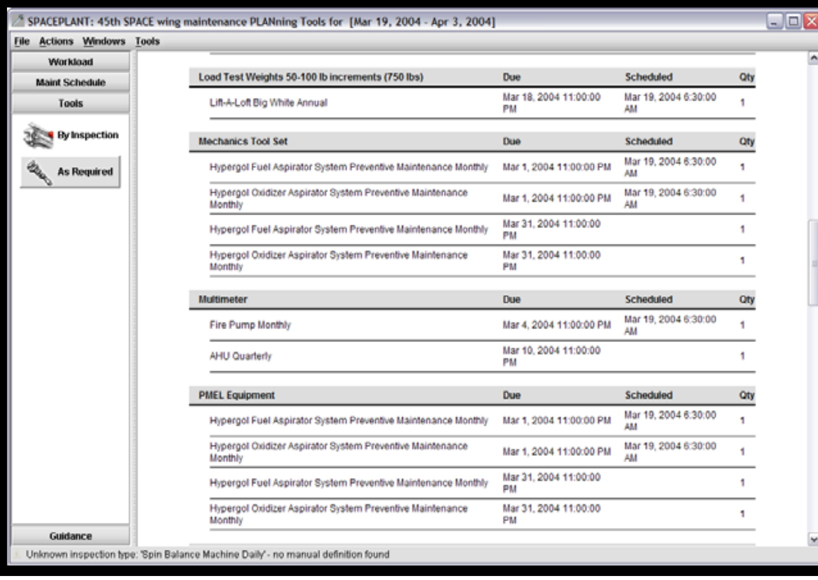

Maintenance Schedule
SPACEPLANT: 45th SPACE wing maintenance PLANning Tool for [Mar 19, 2004 - Apr 3, 2004]
File Actions Windows Tools
Workload
Maint Schedule
72Hr/11Day
Alerts Board
Resource Usage
Tools
Guidance
72Hr/11Day Operations & Maintenance Schedule
Area 59
Date: Fri Mar 19 02:00:00 CST 2004
S.No. Fac. Job# Start End Description Type Mar 2004: 72 Hr
1 55820 Task@1 2004-03-19T06:30:00.000-06:00 2004-03-19T13:00:00.000-06:00 Generic ITAR Placeholder DPF
2 55820 Task@2 2004-03-19T06:30:00.000-06:00 2004-03-26T06:30:00.000-06:00 Lift-A-Loft Big White Annual DPF
3 55820 Task@3 2004-03-22T06:30:00.000-06:00 2004-03-25T07:45:00.000-06:00 15 Ton Underhung Bridge Crane East Bay Monthly DPF
4 55820 Task@4 2004-03-19T06:30:00.000-06:00 2004-03-24T11:15:00.000-06:00 Compressed Air Monthly DPF
5 55820 Task@5 2004-03-19T06:30:00.000-06:00 2004-03-24T06:30:00.000-06:00 Lift-A-Loft Little Blue Semi-Annual DPF
6 55820 Task@6 2004-03-19T06:30:00.000-06:00 2004-03-19T13:00:00.000-06:00 Generic ITAR Placeholder DPF
7 55820 Task@7 2004-03-25T06:30:00.000-06:00 2004-03-30T06:30:00.000-06:00 Lift-A-Loft Little Blue Monthly DPF
8 55820 Task@8 2004-03-22T06:30:00.000- 2004-03-22T06:30:00.000- Breathing Air System DPF
Unknown inspection type: 'Spin Balance Machine Daily' - no manual definition found

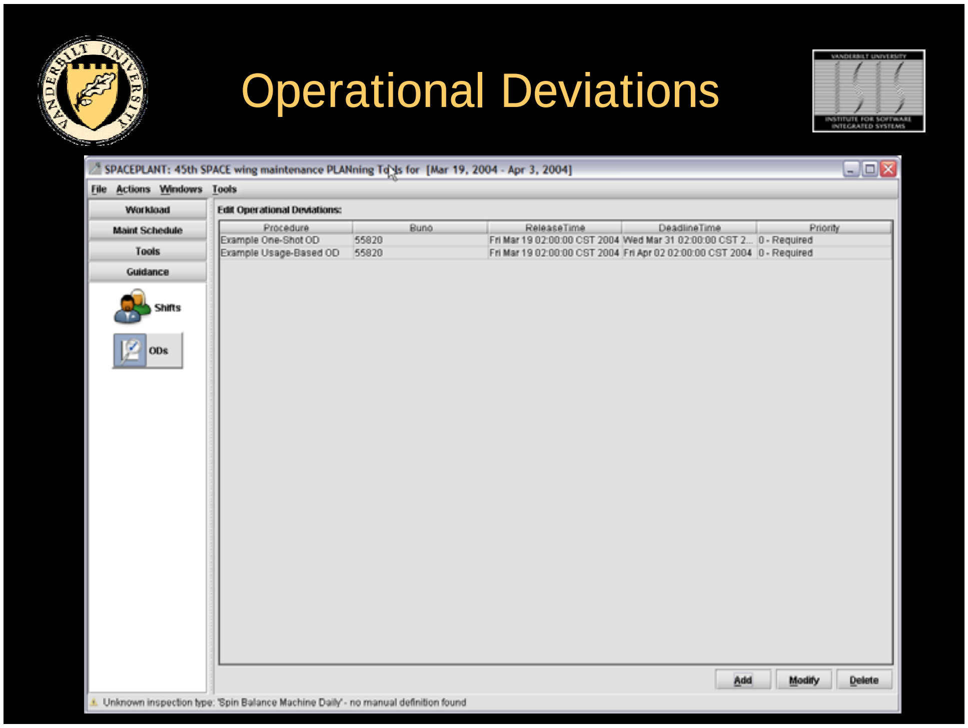


Violations Board
VANDERBILT UNIVERSITY
INSTITUTE FOR SOFTWARE INTEGRATED SYSTEMS
SPACEPLANT: 45th SPACE wing maintenance PLANning Tools for [Mar 19, 2004 - Apr 3, 2004]
File Actions Windows Tools
Workload
Maint Schedule
72Hr/11Day
Alerts Board
Resource Usage
Tools
Guidance
Couldn't successfully schedule these tasks
Facility | Inspection Name | Due Time | Task Name | Ready Time | Deadline | Durat. | Cate.. | Reason
AC-DPF | 5 Ton Underhung Bridge Crane Mai... | 2/19/04:... | 5 Ton Underhung Brid... | 03/19/200... | 03/20/2004... | 1478 | 1 | Due window too narrow
AC-DPF | 15 Ton Underhung Bridge Crane E... | 2/19/04:... | 15 Ton Underhung Bri... | 03/19/200... | 03/20/2004... | 1515 | 1 | Due window too narrow
AC-DPF | Lift-A-Loft Little Blue Monthly | 2/19/04:... | Lift-A-Loft Little Blue M... | 03/19/200... | 03/20/2004... | 1440 | 1 | Due window too narrow
AC-DPF | Lift-A-Loft Big White Monthly | 2/22/04:... | Lift-A-Loft Big White M... | 03/19/200... | 03/23/2004... | 1527 | 1 | Due window too narrow
AC-DPF | Simon Lift Monthly | 2/22/04:... | Simon Lift Monthly | 03/19/200... | 03/23/2004... | 1727 | 1 | Due window too narrow
AC-DPF | AHU Quarterly | 3/10/04:... | AHU Quarterly | 03/19/200... | 04/02/2004... | 6336 | 6 | Optional Job - Due window ...
AC-DPF | Diesel Fuel System Monthly | 3/31/04:... | Diesel Fuel System M... | 03/31/200... | 04/02/2004... | 1440 | 6 | Optional Job - Due window ...
AC-DPF | Hypergol Fuel Aspirator System Pre... | 3/31/04:... | Hypergol Fuel Aspirat... | 03/31/200... | 04/02/2004... | 1440 | 6 | Optional Job - Due window ...
AC-DPF | Hypergol Oxidizer Aspirator System ... | 3/31/04:... | Hypergol Oxidizer Asp... | 03/31/200... | 04/02/2004... | 1440 | 6 | Optional Job - Due window ...
AC-DPF | Diesel Generator Monthly/50 Hour | 3/31/04:... | Diesel Generator Mon... | 03/31/200... | 04/02/2004... | 1440 | 6 | Optional Job - Due window ...
Unknown inspection type: 'Spin Balance Machine Daily' - no manual definition found

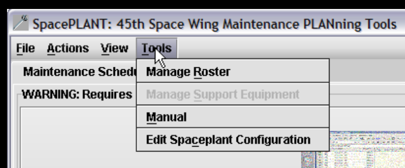

Risk Analysis
SPACEPLANT: 45th SPACE wing maintenance PLANning Tools for [Mar 19, 2004 - Apr 3, 2004]
File Actions Windows Tools
Workload
Maint Schedule
72Hr/11Day
Alerts Board
Resource Usage
Tools
Guidance
Refresh
Reset View
Resources
Time
Projected Scheduled Available
2004-03-25 22:42 Resource:
Marker: 2004-03-24 21:34 Resource: Area Mechanic Satellite
Jump To
Remove
Projected
Scheduled
ACDPF-15 Ton Underhung Bridge Crane East Bay Monthly@3/20/04::23:00@15 Ton Underhung Bridge Crane East Bay Monthly
ACDPF-Lift-A-Loft Big White Monthly@3/23/04::23:00@Lift-A-Loft Big White Monthly
ACDPF-Lift-A-Loft Big White Monthly@3/23/04::23:00@Lift-A-Loft Big White Monthly
ACDPF-Simon Lift Monthly@3/23/04::23:00@Simon Lift Monthly
ACDPF-Simon Lift Monthly@3/23/04::23:00@Simon Lift Monthly
Unknown inspection type: 'Spin Balance Machine Daily' - no manual definition found

Tool Requirements

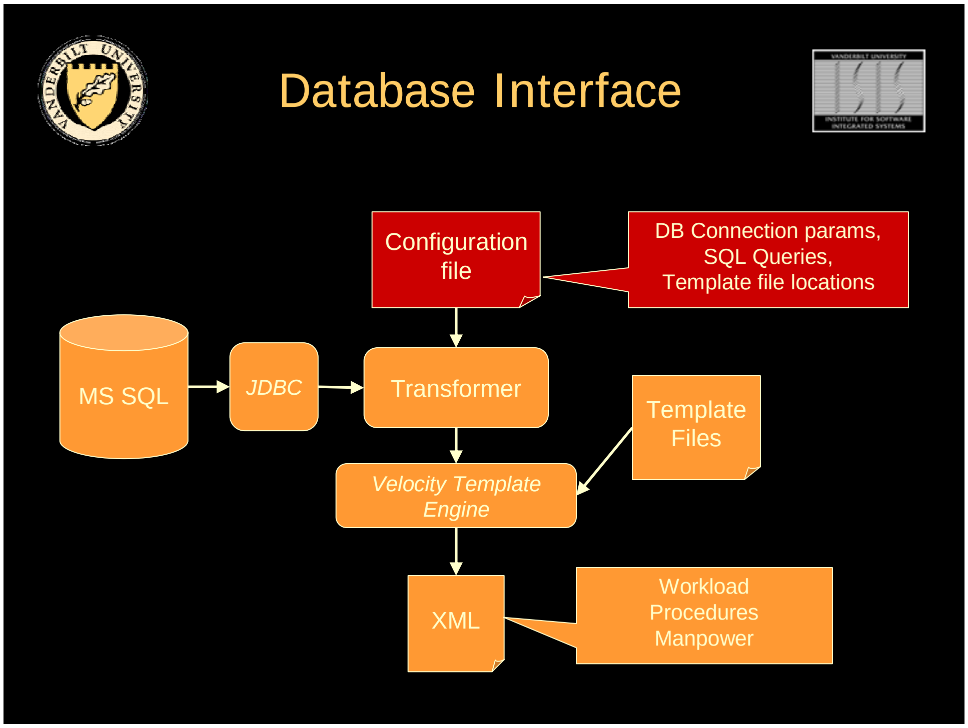



Operational Deviations
SPACEPLANT: 45th SPACE wing maintenance PLANning Tools for [Mar 19, 2004 - Apr 3, 2004]
File Actions Windows Tools
Workload
Maint Schedule
Tools
Guidance
Shifts
ODs
Edit Operational Deviations:
Procedure Buno ReleaseTime DeadlineTime Priority
Example One-Shot OD 55820 Fri Mar 19 02:00:00 CST 2004 Wed Mar 31 02:00:00 CST 2... 0 - Required
Example Usage-Based OD 55820 Fri Mar 19 02:00:00 CST 2004 Fri Apr 02 02:00:00 CST 2004 0 - Required
Add Modify Delete
Unknown inspection type: 'Spin Balance Machine Daily' - no manual definition found

Misc. Tools Menu



- Edit the manpower (e.g. schedule leave for a person).
- Edit the definition of a procedure (or a locally defined *standard operating procedure*).
- Edit the application's configuration settings.


Database Interface
Configuration file
DB Connection params, SQL Queries, Template file locations
MS SQL
JDBC
Transformer
Template Files
Velocity Template Engine
XML
Workload Procedures Manpower